Adaptive Motion Control of Social Robot for Guiding a Users' Group under Dynamic Environment

Bin Zhang

Graduate School of of Informatics and Engineering
The University of Electro-Communications

A thesis submitted for the degree of
Doctor of Philosophy in Engineering

March 2017

Adaptive Motion Control of Social Robot for Guiding a Users' Group under Dynamic Environment

Approved by Supervisory Committee:

Chairperson : Prof. Masahide Kaneko
Member : Prof. Kazuo Tanaka
Member : Prof. Hiroshi Yokoi
Member : Prof. Takayuki Nagai
Member : Associate Prof. Masafumi Uchida

動的環境下でユーザグループの案内を行うソーシャルロボットにおける適応移動制御

張　斌

概　要

近年、ヒューマンロボットインタラクションに関する研究が盛んに行われている。特に案内サービスは人間協調型ロボットに不可欠な機能の一つとして注目されている。案内ロボットに関する従来の研究では、基本的な案内タスクの実現とケースバイケース的な案内機能の追加を中心として努力が続けてこられた。ロボットが人間を案内する際には、ロボットへの追従や相対距離の維持などユーザ側からの協力が必要だった。これに対し本論文では、ユーザの動きに適応した柔軟な案内サービスを提供できるソーシャル案内ロボットの実現を目指している。ソーシャル案内ロボットとはユーザの動きを制約せず、その動きに適応しながら同行を行うことによって案内タスクを達成させるロボットである。このために本論文では、環境理解機能、ユーザグループ追跡機能とユーザに適応した移動制御モデルをロボットに搭載した。

本論文は以下の５章より構成される。

第１章は序論である。本論文における研究の背景と先行研究について説明し、その現状と課題をまとめた後、本論文の目的を述べる。

第２章では、ロボットの移動経路の自動生成方法について述べる。まず、案内ロボットが働く環境の２次元地図を SLAM(Simultaneous Localization and Mapping) 手法によって作成する。この際、LRF によって得られる２次元情報と Kinect によって得られる３次元情報を合わせて、よりロバストな地図を効率良く生成する。Kinect から取得した３次元情報によって人物や壁、机、椅子などの物体を認識し、壁のようなものが存在する不動領域を地図上に早めに反映させる。一方、人物が存在する動領域を地図上から削除する。次に、ロボットの移動経路の生成は大局的手法と局所的手法を統合して行う。ダイクストラ法に基づいて最終目的地までの最短経路を生成する。この経路は随時更新される。その後、ポテンシャル場法を用いて、生成した経路からサンプリングしたサブゴールに向かってロボットを移動させる。この統合手法によって動的な環境にも対応でき、また、最短経路での案内が可能となった。

第３章では、案内の対象となる複数人のユーザグループに対し、ロバストな行動検出と追跡を行うシステムについて述べる。ユーザグループの追跡は、MCMC (Marcov Chain Monte Carlo) パーティクルフィルタを用いて実現する。ロボットに搭載した Kinect を利用して各ユーザの顔、色などの個人情報を取得して統合し、個人認識を行う。その結果をパーティクルフィルタの観測尤度として利用し、他の歩行者と区別しながら、複数人のユーザグループの追跡を行う。各ユーザをそれぞれ独立したパーティクルグループで追跡し、追跡結果を近くにある人物領域に近似する。ただし、各人物領域は複数のユーザ追跡結果として共有可能にする。本人物追跡システムの有用性を実験室環境下で様々な人数から構成された複数のユーザグループの追跡実験を通じて評価した。また、本追跡手法を保育支援システムのための子ども行動追跡へも応用した。以上により、２０人程度までの集団が室内で行動するような複雑なシーンに対してもロバストな行動追跡を行えることを明らかにした。また、途中で追跡に失敗しても、その後ユーザの再検出を行うことによって追跡の続行が可能であるという特長がある。

第４章では、ユーザグループの動きに適応した移動制御フレームワークを構築する。ユーザグループとサブゴールにそれぞれ特別なポテンシャル場を生成し、他の歩行者や障害物から生成した斥力ポテンシャル場と統合する。統合ポテンシャル場基づいてロボットを制御し、ポテンシャルが最も低くなる方向にロボットを移動させることで、ロボットはユーザの動きに適応しながらソーシャル的な案内サービスを提供する。ユーザグループの移動速度を予測し、その速度に合わせてロボットが移動する。ユーザが一時的に経路から離れ、後ほど最初の案内タスクに戻る様な場合であってもロボットは案内モードと追従モードを自動的に切替えて対応する。ただし、ロボットが追従モードで動く場合であってもユーザグループに追従するだけではなく、常に最終目的地を考慮し、案内の再開に対応しやすい態勢で待機しながら、ユーザグループの隣でついていく。ユーザグループが元の案内タスクに戻ってくると、ロボットは短い反応時間で、無駄な経路を通らず、スムーズに案内を再開することができる。ロボットが常にユーザの傍に待機することで、より円滑な案内サービスを実現する。シミュレーションと様々なユーザグループの動きに適応した案内実験によって、本適応的移動制御手法の有効性を明らかにした。

第５章では、本論文のまとめと今後の課題について述べる。すなわち、本

論文ではユーザグループの動きに適応し、ソーシャル的な案内サービスを提供するロボットを提案した。基本的な案内タスクの実現を中心とした既存研究に対し、提案手法では柔軟な案内の実施を重視し、ロボットへの追従や相対距離の維持といったユーザからの協力を必要としない。ロボット側がユーザの行動を理解し、その動きに適応する。提案手法に基づいてロボットが環境を理解し、ポテンシャル場を用いて適応的な動きを自動的に生成できることをシミュレーションや実機実験で検証し有用性を明らかにした。今後の課題としては公共的な場所での動作検証が挙げられる。また、ユーザの心的な状態を推定し、ユーザの意図に合わせる様な案内サービスに関する検討が挙げられる。

Adaptive Motion Control of Social Robot for Guiding a Users' Group under Dynamic Environment

Bin Zhang

Abstract

In recent years, human-robot interaction has become one of the fastest growing research fields. One important function of service robots used in different fields is to guide users from one place to another. Conventional researches about guide robot focus on realizing the guide task and adding the functions of the guide robot by changing the motion states of the robot case by case. The guide robot requires the users to follow it or maintain a fixed distance from it. In this thesis, the purpose is to make a social guide robot that can provide adaptive social guide service to the users. Social guide robot means that the robot needs to adapt to users' activities and always accompany with them during the guiding process instead of limiting the users' motions. To realize this purpose, the environment understanding function, users group tracking function and adaptive motion control model are added to the guide robot.

This thesis is organized as follows.

Chapter 1 is the introduction part. It introduced the background and the conventional researches about autonomous guide robots first. After explaining the problems and tasks about the conventional researches, the purpose of this thesis is introduced.

Chapter 2 explained, in detail, the path planning method for the guide robot. A 2D map of the work environment is generated in advance by Simultaneous Localization and Mapping (SLAM) method, which is usually used to generate a 2D map. The 2D map is generated by combining 3D information from Kinect sensor and 2D information from LRF sensor on the robot. With 3D information from Kinect sensor, humans and objects like walls, desks or chairs can be recognized. By reflecting the immobile areas like walls on

the map quickly and deleting the mobile areas like humans from the map, a more robust map for path planning is generated. The path planning part is based on combing the global and local path planning methods. Dijkstra's Algorithm is used for global path planning and the shortest path is generated and updated online. Potential field method is used for local path planning by moving towards the sub-goals, which are sampled from the shortest path. In this way, the guide robot can deal with dynamic environment and guide the users in the shortest way.

Chapter 3 explained the way to recognize and track the particular users' group when the robot conducts the guide task. The problem is solved by using MCMC particle filter. Each of the users is detected and recognized by the face and color information. The recognition result is used as the observation likelihood in the tracking process. Each of the users is tracked by an independent group of particles and the tracking results are modified to the nearest human area. All of the human areas are allowed to be shared by multiple users since people may be closed with each other and form big areas containing multiple users. The effectiveness of this method was proved through tracking different groups of users under lab environment, and furthermore it is successfully applied to continuous monitoring of multiple children in a nursery school.

Chapter 4 introduced the proposed framework to control the robot adaptively. In this framework, special artificial potential fields for the users and the sub-goals are generated separately, and integrated with that generated from the obstacles in the environment. The robot is controlled by the integrated potential filed and moving towards the point with the lowest potential. In this way, the guide robot can adapt to the users' activities and provide sociable tour guide services. The robot predicts the moving speed of the users' group and adapts to it to maintain the social distance. Moreover, with the proposed framework, users can deviate from the guided path temporarily and return to the original task afterward. Instead of waiting for the users and taking the risk of losing them, the robot deviates from its original path to follow the users while preparing for returning to the guiding task. The robot restarts the guiding task from the best posture that ensures the mode changing process smoothly. The adaptive motions can be automatically generated by the proposed framework. The guide robot controlled by the proposed framework can always accompany with the users during the guide tasks and provide them guide services. The effectiveness of the proposed framework was demonstrated by simulations, and the users were guided to their destination in a sociable way by the robot in the experiments, in which different groups of users with different numbers were guided by the robot.

Chapter 5 described the conclusion of this research. In this thesis, a social guide robot is designed by adapting the motions of the users group and accompanying them during the guide task. Different from the conventional works that were focused on realizing the guide task, this work tried to improve the quality of the guide task, allowing the users move more freely without any restrains. The sociable guide robot should not ask the users to follow it and maintain a proper distance with it. Instead, the robot should understand the will of the users and adapt to their motions. It is realized by environment understanding and controlling the robot by integrated potential field method, in which special potentials are generated from sub-goal and users. From the experimental results, the robot can adapt to the users' motions. Moreover, future works of this research is discussed before applying to real environments. Verification experiments in public places are needed. The guide robot can be more considerate by analyzing the mental states of the users to predict intentions of the users.

Contents

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
ICP	Iterative Closest Point
LRF	Laser Range Finder
SLAM	Simultaneous Localization And Mapping
MCMC	Markov Chain Monte Carlo
PF	Particle Filter
PCL	Point Cloud Library
RSS	Residual Sum of Squares
MAP	Maximum A Posteriori

Chapter 1

Introduction

1.1 Background

When people come to a new place, they may feel confused, lost or worried. This is a common physiological response as a human being because they are unsure where to go. That is why a guide is usually needed for the tourists. People usually form a group when they visit the same place or when they want to move together. Tour guide is a occupation, that is guiding tour, letting the visitors have a good experience about the place and solving the problems that may occur during the tour [1]∼ [4]. Most of museums, visiting spots and factories provide professional tour guide services, for impressing the visitors and maintain the order. A good tour guide will make people feel better and want to come to the places again. However, the cost is huge to train a professional guide, as he/she needs to have rich knowledge to improve the guide service. The tour guide knowledge includes language knowledge, the history of cultural knowledge, knowledge of policies and regulations and psychology and aesthetics knowledge. Moreover, in the developed countries like Japan, aging and low birthrate have become great social problems, which leads to the shortage of labors. Less and less people want to enter the service industry year by year. It is highly necessary to solve this problem by technical skills. Researchers have developed the tour guide robots [5] to solve this problem. Nowadays guide robot would simply guide people through the place they are visiting and explain the surrounding for the visitors. They believe that tour guide robots can replace the human guides in the future.

With the development of robotics and artificial intelligence (AI), more and more researchers came to this field and proposed more smart guide robots [5], as shown in Fig. 1.1. However, most of the robots were focused on how to complete the guide task, but the quality of the guide service is seldom considered. Qualified service is an important part as a guide robot. Tour guides should understand the visitors' intentions and try to give no limits to them [2] when they move around. That is to say, the tour guide should adapt the motion of the visitors and provide them services that they expects most. The situations are the same with guide robots. Visitors expect the guide robot to be more sociable. Most of the conventional researches focused on realizing the guide task, but they usually need the cooperation of the users. The users need to follow the robot and keep the relative distance with it. Even some robots can follow the users when they deviate their original path [70], the robot realized the functions by designing kinds of motion rules case by case. The guide robot cannot expect that the users will always follow it or maintain a fixed distance from it; instead, it needs to adapt to user activities and always accompany with them during the guiding process. Instead of the methods in the conventional researches, which focus on realizing the adaptive motions by changing motion states of a robot case by case, a general adaptive motion control framework is needed. To the best of my knowledge, no researcher has proposed a general framework by which all these adaptive motions of an autonomous mobile robot can be generated at the same time. The purpose of this research is to make a social guide robot for serving a user's group under dynamic environment. In the future, it is expected that this kind of robots can provide qualified guide service instead of professional human guides.

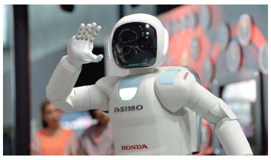
1.1.1 What is a social guide robot?

Generally speaking, the word of "social" has a wide range of meanings even as a guide robot. All kinds of behaviors by which the robot makes users feel comfortable can be considered as social motions. For example, some researches make the robot have interactions with the users during the guide

(a)Rhino[6][7]

(b)Minerva[8]

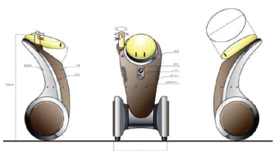
(c)Robovie[9]



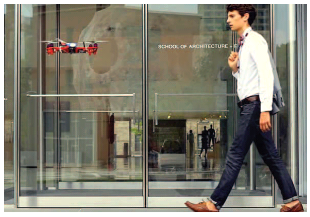
(d)Asimo[10]

(e)Tawabo[11]

(f)Toyota robot [12]



(g)Ride robot[13]



(h)Skycall[14]

Figure 1.1: Guide robots.

task, or talk politely with them. Some make the robot judge the intention of the user and communicate with the user who wants to talk or ask for help. Here, in this dissertation, “social” is defined as the adaptive motion of the robot. When the robot guides the users to their destination, it cannot require the users paying attention to the relationship and keeping a proper distance. It is the robot that needs to pay attention to the users, and adjusts its motion to best adapt to the intention of the users. When the users want to accelerate, the robot needs to move faster, and when the users stop, the robot needs to wait for the users. Moreover, the users are allowed to move around in their free will during the guide task. They may be interested in anything nearby, and have a look at those things before returning to the original tasks. These "dropping by" behaviors frequently happen for foreigners, children and the people who are visiting the places that they are interested in. For example, when a group of foreigners comes to Japan for the first time, a guide robot can be used to guide them to the train station or shuttle bus stops in the airport. However, any kind of Japanese style things may cause their attention, which will lead them to drop by before leaving. As a social robot, it cannot force the visitors to leave, but adjust its speed and path to make the users enjoy their time. They should be free to move and the robot needs to accompany them during the task. After finishing their sight viewing, the robot needs to be ready to restart the original task and guide them to the train stations or bus stops. These adaptive motion controls of the guide robot are required to perform the good manners of Japanese services. This dissertation focuses on solving these kinds of problems, and a guide robot, which can adapt its motion to the users and accompany them well, besides completing the guide task, will be evaluated as a social guide robot. In terms of motion control, a sociable guide robot needs to carry out the following four functions: it must (1) adjust its speed to match that of the user (e.g., young people may move faster than elderly, or the same group of people may change their speed depending on their interests); (2) maintain its relationship by maintaining the social distance with the users (e.g., by following them if they deviate from the guiding path); (3) prepare to restart the guiding task during "Follow" mode; and (4) take the users to their destination by the shortest path.

1.1.2 What is the dynamic environment ?

In order to provide guide services to the users, the robot needs to understand the work environment first. Environment understanding contains two parts: one is mapping the place where the robot will work in advance, and the other is detecting and tracking the users' status, besides other pedestrians and obstacles in the environment when the robot conducts the guide work. Considering the practicality of the guide robot, it should be able to work under real public environments like museums or airports. Here, the "dynamic environment" means that there are kinds of obstacles and other pedestrians in the environment, and the user group contains a group of people. The map of the work environment can be generated in advance, but the robot still needs to recognize the mobile obstacles and pedestrians in the space to prevent collisions when moving around. Laser range sensors are usually used for detecting and locating the obstacles. The obstacles in the real environment contain not only that are easy detect by laser range sensors, like walls or boxes, but also that are hard to detect by laser range sensors like desks or chairs. As shown in Fig. 1.2, the robot can detect the obstacles (like Fig.1.2 (a)) and avoid collisions with them easily, but the desks and chairs (like Fig. 1.2 (b)) are hard to be totally recognized. For example, the desks are only shown as 4 points on the map because only their legs are detected by the laser range sensor. The plane part of the desk is hard to be detected and the robot may judge that it can get through the space among the legs when it chooses its path. In this case, collisions would happen. The guide robot needs to renew the map with 3D information for avoiding these situations. Secondly, the pedestrians around are free to move in the environment and the robot also needs to avoid colliding with them. Different from the real obstacles, the pedestrians are usually not registered on the map as they may move away any time. These kinds of moving objects are needed to be considered (like Fig. 1.2 (c)). The group of the people should be deleted in the map generated in advance for reflecting the real environment. This work can be realized by fixing the map manually before the guide task. Thirdly, the robot needs to guide a group of people. Here, a user group within 20 people is considered considering the practicality of the guide robot. General sensors like Kinect sensor are released with human tracking libraries, but they can only deal with a small number of people (for Kinect, under 6 people) [5]. Multi-sensor systems can detect a big number of people, but they need many sensors setting in the environment [6]. The guide robot needs to track a group of people with its own sensors to make sure its flexibility and adaptability under different environments.

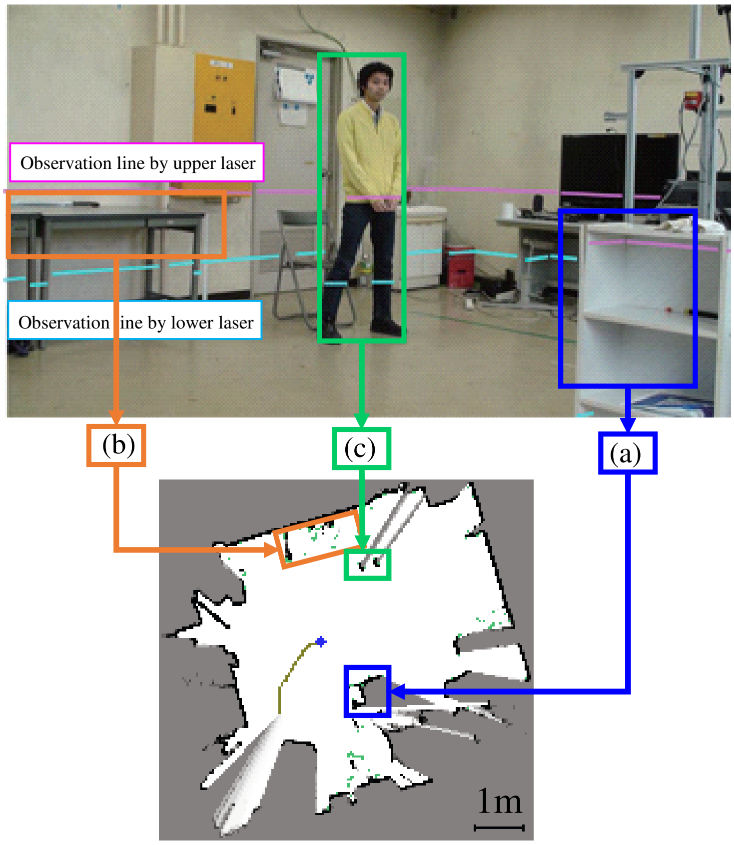


Figure 1.2: Detection and mapping results by laser range sensors.

1.1.3 Main Objectives

With an increasing number of services provided by guide robots, it is crucial to ensure service quality. Users expect the guide robot to be more sociable. However, few researchers are working on how to make the robot more sociable to improve the quality of guiding service. The guide robot cannot expect that the users will always follow it or maintain a fixed distance from it; instead, it needs to adapt to user activities and always accompany with them during the guiding process. In this thesis, a framework is proposed to control the motion of the robot to accompany the users and provide sociable guiding services. In terms of motion control, a sociable guide robot needs to carry out the following four functions: it must (1) adjust its speed to match that of the user (e.g., young people may move faster than elderly, or the same group of people may change their speed depending on their interests); (2) maintain its relationship by maintaining the social distance with the users (e.g., by following them if they deviate from the guiding path); (3) prepare to restart the guiding task during "Follow" mode; and (4) take the users to their destination by the shortest path. Conventional research focuses on realizing one or more of these by changing the motion states of the robot case by case. However, the number of rules for this case-by-case method may diverge quickly when the environment becomes more complex. To the best of my knowledge, no researcher has proposed a general framework by which all these adaptive motions of an autonomous mobile robot can be generated at the same time. Thus, a framework is designed to control the robot to move adaptively in a socially acceptable way. The guiding quality is improved by accompanying the users with adaptive motion control, using a single general framework rather than multiple case-by-case rules. With the proposed framework, the adaptive motions are formed automatically, and all four above-mentioned functions for a sociable guide robot can be realized. By considering the situations of the users and real time updated subgoal separately, the proposed framework can control the robot in such a way as to adapt to the users' activities and the users can move without any restrictions.

1.1.4 Related Researches

Tour guide robot has been researched for years, and a series of guide robots are released. Fig. 1.1 shows some guide robots for realizing a tour guide robot. Table 1.1 shows the advantages and disadvantages of the conventional guide robots. Fig. 1.1 (a) shows a guide robot called Rhino. It is used as a tour guide robot [6] [7]. This robot was shown to work in Deutsches Museum Bonn in Germany. Rhino uses the given map before it starts to work and moves in the directions that were set previously. Moreover, a website is provided, by which the staffs can locate the position of the robot and to manipulate the robot at certain times. Fig. 1.1 (b) shows another guide robot called Minerva [8]. The robot can generate the map around it when it shows the users around. Meanwhile, it can be controlled from a website and

Table 1.1: The advantages and disadvantages of conventional guide robots

robot	advantages	disadvantages
Rhino [6] [7]	motions set; website to localization and manipulate	previous map needed; unadjustable motions
Minerva [8]	mapping; website control	motions adjusted by humans
Robovie [9]	smart; cute; programmable	expensive
Asimo [10]	humanoid; interactions	environment sensors by wireless networks
Tawabo [11]	smiling faces; multiple languages	few sensors; difficult to control
Toyota robot [12]	gestures; emotions; tag recognition; group guide	require users' cooperations
Ride robot [13]	ridable; user control	manual control few sensors
Skycall [14]	GPS navigation; multi-sensors; work under any road conditions	hard to control; less security sense

the video information taken from the sensor of the robot can also be shown on the website in real time. The staff can control the robot by watching around in the robot view. Fig. 1.1 (c) shows a humanoid robot called Robovie. This figure shows the third version of the robot, which seems more flexible and cute than the previous versions. It contains laser and camera sensors to scan the environment around, and it can be programed to move automatically through the wheel motion platform. Some researches used it as a guide and do presentations for the users [9]. Fig. 1.1 (d) shows the guide robot Asimo, which was developed by Honda [10]. This humanoid robot can do kinds of interactions with the users, and it locates itself by using the sensor setting on the ceiling of the room that the robot works. The robot is designed just like a human being, and it connects with its sensors by wireless networks, which make it seem more flexibly and intelligent. It is also applied to the real environment in a museum in Tokyo. When it interacts with users, the robot can recognize the guy who is putting up his hands. However, the robot cannot distinguish the other lift-hand motions like taking a photo, etc. The robot also uses a screen to communicate with the users. Fig. 1.1 (e) shows a robot called Tawabo [11], which works at the Tokyo tower. This robot has a smiling face on it for most of the time to make users comfortable, and it can speak 4 different kinds of languages, which impressed many foreigners. The screen on the robot shows the information that the robot wants to express. Fig. 1.1 (f) shows the famous Toyota robot developed for guide work [12]. This robot is exhibited at the Toyota Kaikan Exhibition Hall, Tokyo. It can give different gestures and emotions during the guide task, which make it more similar with human beings. It can communicate with a particular person by recognizing one's name tag. This robot guides huge number of people by moving slowly. Fig. 1.1 (g) shows a kind of different kind of guide robot, which has two wheels and allows a user to ride on it [13]. The user can control the robot and they are provided a tablet PC to enjoy the tour. Fig. 1.1 (h) shows a flying guide robot called skycall [14]. The GPS system is used for the robot for navigation and it successfully guide the people in a campus. The users communicate with the robot by an application software. Two cameras on the robot help it to detect the objects around it.

However, all of the developed guide robots were concentrated on how to finish the task, and how to provide communication tools for the users. To the best of my knowledge, this is the first time that a single framework is proposed to adaptively control the motion of the robot, for constantly accompanying the users during the guide task. The robot needs to respect the will of the users and provide considerate guide services.

1.2 Contribution of the Thesis

In order to make a social guide robot, a framework is designed to control the robot to move adaptively when guiding a users' group in a social acceptable way under dynamic environment. The robot needs to understand the dynamic environment, and judge the status of the users' group to provide information for the guide framework.

1.2.1 Mapping and Path Planning for Mobile Robot

For the part of mapping and path panning for mobile robot, this thesis makes efforts to solve the following two problems:

- The immobile area grid map based SLAM method for generating 2D grid map cannot deal with 3D obstacles like desks or chairs.
- Global path planning methods cannot deal with dynamic environment, and local path planning methods cannot make sure the generated path is globally optimum.

The conventional studies about mapping and path planning for mobile robot have been surveyed. The immobile area grid map based SLAM method is applied to the guide robot to generate a 2D map, which is easier for path planning than 3D map. However, the conventional method [35] used laser range sensor information to recognize objects and generate the map, it cannot deal with 3D obstacles. This method is extended by combing 2D and 3D distance information to detect obstacles in space. The objects like walls, desks, chairs and humans are recognized with higher accuracy, and all kinds of potential mobile objects are deleted from the map. The generated 2D map reflects the real environment better.

Moreover, Dijkstra's Algorithm based global path planning and potential field based local path planning methods have been used together to make the robot move flexibly by the shortest path. The global shortest path is generated and updated online so that the robot can always move by the global optimum path. The potential field method is used to make sure the robot can deal with dynamic environment. This method is easy to implement and suitable to develop the adaptive motion control framework.

1.2.2 Simultaneous People Recognition and Tracking System

For the part of simultaneous people recognition and tracking system, this thesis makes efforts to solve the following two problems:

- The conventional tracking systems [47] [46] usually fails to track people when they always go across with each other, thus the application for long term tracking under crowed environment is limited.
- Once the system starts to track a wrong person, it is hard to fix the error by the system.

In this study, a general people recognition and tracking system is proposed. With this system, the particular people can be continuously recognized and tracked even under crowed environment. The recognition algorithm keeps recognizing the detected people to improve the tracking accuracy, and the tracking algorithm records the motion history of the people and predicts the positions of them to improve the recognition accuracy. Compared with conventional methods [49], the system has higher tracking accuracy and it can automatically modify the errors that are caused by occlusions. That is to say, even if the system failed to track the right person, it will be modified when the particular user is recognized again.

1.2.3 Adaptive Motion Control Framework

For the part of adaptive motion control framework, this thesis makes efforts to solve the following two problems:

- The conventional guide robots [61–64] require the users to be cooperative to finish the guide tasks.
- In the conventional methods [70] [69] [67], the robot is controlled by case-by-case rules to form adaptive motions.

A framework of adaptive motion control for social guide robot is proposed. With this single framework, two kinds of adaptive motion modes can be formed automatically and switched between them according to the need of the users. This framework focuses on improving the guide services by adapting to the users' motions without any limitations and cooperation from them. With this framework, the guide robot can (1) adjust its speed to match that of the user (e.g., young people may move faster than elderly, or the same group of people may change their speed depending on their interests); (2) maintain its relationship by maintaining the social distance with the users (e.g., by following them if they deviate from the guiding path); (3) prepare to restart the guiding task during "Follow" mode; and (4) take the users to their destination by the shortest path.

1.3 Outline of the Thesis

The contents of each chapter and their relationships are shown in Fig. 1.3. The rest of the thesis is organized as follows:

Chapter 2: It is explained, in detail, the way to generate a robust map for understanding the environment where the robot will work. A method of combining 3D information for generating the 2D map is used based on SLAM. The method of combing global and local path planning methods is used to generate the path for the guide robot. The shortest path on the global map is generated first to make sure the path is globally optimum. Then the robot is controlled by the potential field method to avoid collisions with the obstacles. The points sampled from the generated paths are used as subgoals to control the robot.

Chapter 3: The system to recognize and track the particular users when the robot conducts the guide task is proposed. The problem is solved by using MCMC particle filter, and the MCMC particle filter method is extended the conventional method by modifying the tracing results according to human detection results and allowing multiple users share one detected area. The effectiveness of this method was proved through tracking different groups of users under lab environment, and it is successfully applied to continuous monitoring of multiple children in a

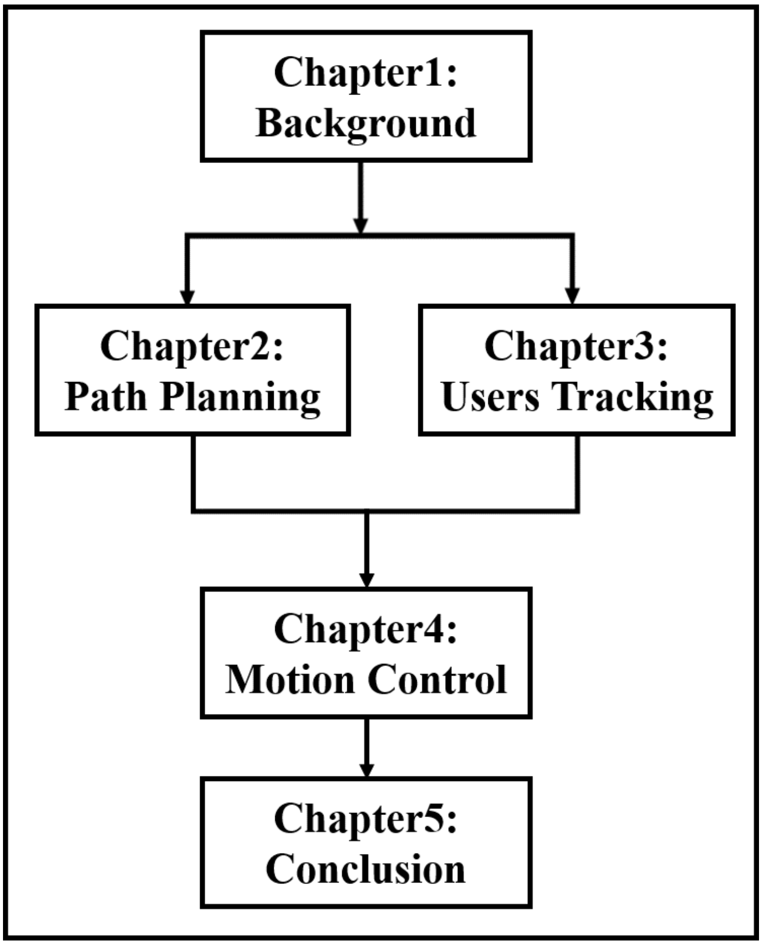


Figure 1.3: The relationship among different chapters.

nursery school.

Chapter 4: The proposed framework to control the robot is explained. In this framework, special artificial potential fields for the users and the goal are generated separately, and integrated with the potential that is generated from the obstacles in the environment. The effectiveness of the proposed framework was demonstrated by simulations, and the users were guided to their destination in a sociable way by the robot in the experiments, in which different groups of users with different numbers were guided by the robot.

Chapter 5: Conclusions and future work are given. A framework is designed and proved effective to control the robot to move adaptively in a socially acceptable way. The system can be improved in the future by adapting to real public environments and analyzing and predicting the motions of the users.

Chapter 2

Path Planning Under Dynamic Environment

2.1 Introduction

For any kind of mobile robot, navigation system is one of the most important technology, and path planning is an essential part for navigation. Global path planning method is usually used when the environmental model is available. Researchers tried to divide the work space into many grids with location information [15]. The gird map method is good at finding the shortest path and representing the environment. But it has the problem of huge amount of calculation when the environment scale is big. Some researches have tried to improve the efficiency by using octree grid method [16]. However, these kinds of methods still have the problem that the generated paths may be very close to obstacles, and they cannot deal with dynamic environments, like mobile objects. For the case that the environment is partly known or unknown, local path planning method is usually used. The potential field method is a famous method for local path planning, and it is flexible enough to deal with mobile objects as this method is based on the information gotten on-line by sensors [17]. The weakness of this method is the local minimum problem. Researchers tried to solve the problem by using hybrid potential field method [18] or mode changing with Laplace method [71]. A^* method also improved the speed of path planning [19]. All of the methods mentioned above have a common problem that they treat the robot and other obstacles as points. Their shape are not considered and the space between the robot and the obstacles may trend to be too small, which will lead to collisions easily. In this chapter, the method of combining global and local path planning methods is used to generate the path, and the points sampled from the generated paths are used as subgoals to control the robot.

2.2 Environment Model: SLAM

2.2.1 Introduction

In recent years, Simultaneous Localization and Mapping (SLAM) has become one of the most fascinating research fields [20] [21]. Localization and mapping are two essential information for any autonomous mobile robot working in human living environment. As the basic technique for path planning and activity planning, the generated map can also be used for robot navigation later in the same environment [22]. Laser scanners have been widely used for SLAM [23] [24] as they usually have high accuracy to detect distances in front and have a wide view angle. For example, Wang *et al.* [23] successfully realized SLAM under crowed environment and generated a map without moving objects. However, conventional methods still have a few problems to be solved. Firstly, the accuracy of localization needs to be improved. The odometry information gotten from the mobile robot can be used for localization. Whereas, its accumulative error caused by the differences between the two wheels of the robot, rode condition or sensor noise will be fatal with time going on. Iterative closest point (ICP) [25] method is usually used for localization in SLAM process by matching the point cloud gotten from the laser sensor with that in the global map. However, there may be many matched places in the map. Especially under familiar environment like the corridor, the point clouds may be matched everywhere. A localization method by integrating these two kinds of information is proposed. The accuracy of localization is greatly improved under familiar environment. Secondly, the generated map based on laser scanner sensor only contains objects information from the plane where the sensor is located. It is hard to provide measurement in 3D environment where objects are in different heights. For example, desks and chairs are normal in indoor environment, but a 2D scanner sensor may only detect their legs as objects, which turn to four points on the generated map. Collision may happen during autonomous navigation for the robot. This problems can be solved by using multiple 2D sensors or 3D sensors [26] [27] [28]. Unfortunately, the accuracy of multiple 2D sensor is still limited by the number of sensors and the cost of multiple 2D sensors or 3D sensors with wide view angle. Recently, 3D sensors with low cost like Microsoft Kinect sensor have been released but the view angle is still narrow and the information is insufficient for robot localization. Research in [29] tried to rotate a 2D scanner sensor by a motor to detect 3D information, and the real time performance is limited. In this chapter, SLAM is realized by integrating the information of a laser scanner sensor called Laser Range Finder (LRF) and a Kinect 3D sensor. The idea of sensor placement is familiar with Lin *et al.* [30], but different from their work, the generated map is more robust in removing potential moving objects. Thirdly, most of conventional researches [31] [39] [42] [34] tried to remove the moving objects from the generated map, but they are difficult to deal with potential moving objects that are static at the moment but actually able to move in the future. Ito *et al.* [35] tried to deal with this problem by using an immobile area grid map, in which a step of describing the degree of use of the observed object in the map by probability is added. The immobile area grid map based SLAM method is used for localization and mapping for autonomous mobile robot.

2.2.2 Robot Self-Localization

To guide the users correctly, the robot needs to locate itself accurately. The iterative closest point (ICP) method can be used for localization [25]. Its working process is shown in Fig. 2.1. The newly detected distance data by the laser sensor is matched with the model data, as shown in Fig. 2.1 (a). These two types of data should be similar throughout continuous frames, and the location with the closest match can be determined by minimizing the distance between corresponding points, as shown in Fig. 2.1 (b). The calculated positional relation shows the motion of the robot, and the localization of the robot can also be calculated as shown in Fig. 2.1 (c). However, this method has a problem that there may be multiple matched places with the map. For example, in the corridor, the model data are similar with the new detected sensor and matched with the detected distance data everywhere, as shown in Fig. 2.2. The odometry information received from the mobile robot can also be used for localization. Notably, the accumulative error caused by the differences between the two wheels of the robot, road condition, or sensor noise will be fatal as time goes on. Thus, these two types of information are integrated by using the Markov Chain Monte Carlo [36] particle filter method. During the prediction step, the motion information obtained from odometry is used to predict the position of the robot, mixed with Gaussian noise in order to use it as the motion model. During the evaluation step, the prediction results were evaluated and re-weighted according to the matching rates between the detected distance data and the model data, same with the ICP method. The robot position then can be calculated as the re-weighted mean of all prediction results.

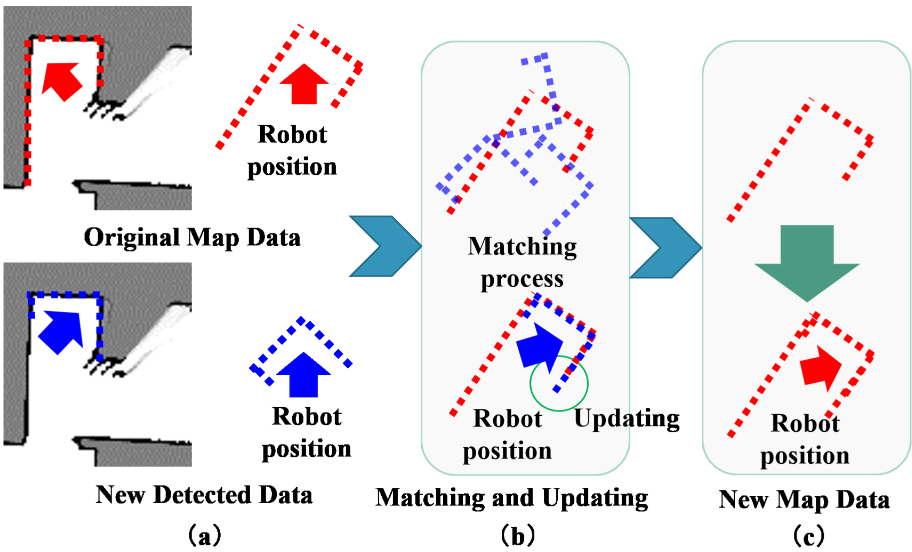


Figure 2.1: Working process of the iterative closest point (ICP) method.

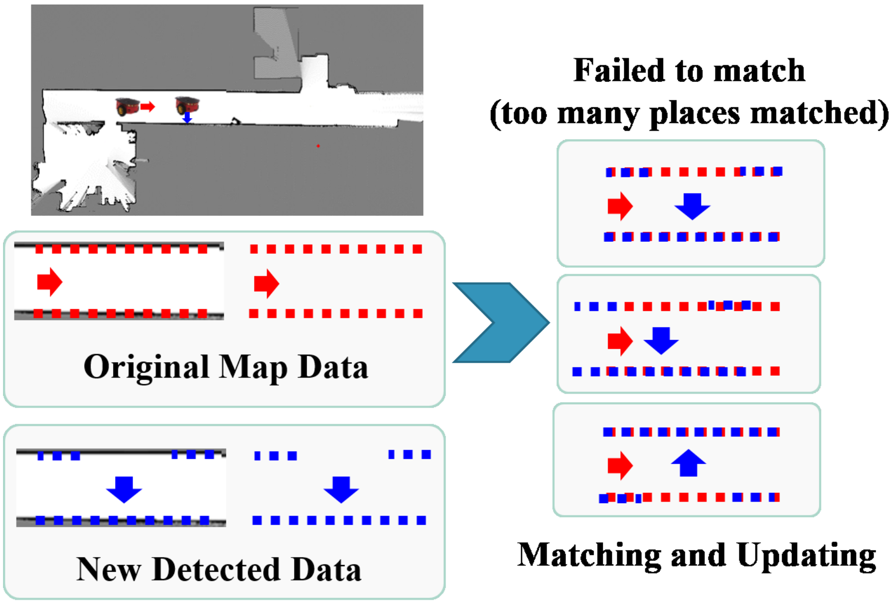


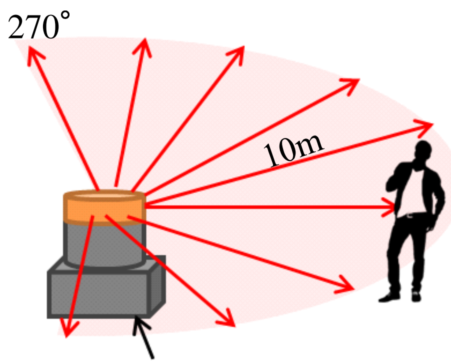
Figure 2.2: Matching in multiple places when using the iterative closest point (ICP) method.

2.2.3 Integration of 2D and 3D Information

To understand the environment around the robot, distance sensors are usually used for SLAM. In the proposed system, the Laser Range Finder (LRF) sensor is used for getting the 2D distance information, and Kinect sensor is used for getting the 3D information around the robot. LRF sensor is chosen as the UTM-30LX made by Hokuyo Co.. It can sense the range of 270° in front of the robot with the maximum distance of 10m. The degree step is 0.25°. The LRF sensor is shown as Fig. 2.3 (a) and its sensing range is shown as Fig. 2.3 (b). Kinect sensor can get the color and depth information at the same time in front of the sensor. It is made by Microsoft Co. and the Kinect V2 is chosen for the experiment. The sensing range of the Kinect sensor is 70° in horizontal and 60° in vertical with the maximum distance of 8m. The color image resolution of the Kinect sensor is 1920*1080 pixels. The Kinect sensor is shown as Fig. 2.4 (a) and its sensing range is shown as Fig. 2.4 (b).

LRF has a large sensing range but it can only detect the distance informa-

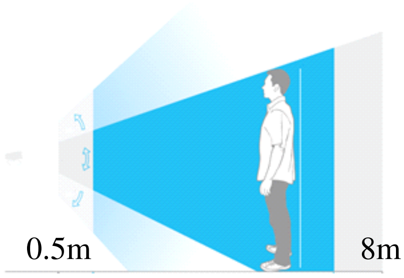
(a) LRF sensor



(b) Sensing range of LRF

Figure 2.3: Introduction of LRF sensor.

(a) Kinect sensor



(b) Sensing range of Kinect

Figure 2.4: Introduction of Kinect sensor.

tion in a horizontal plane. The objects in front of the robot can be detected well on the sensing plane. For the scene shown in Fig. 2.5 (a), the object detection result is shown in Fig. 2.5 (b). Kinect can detect 3D distance information and color information at the same time, but its sensing range is limited. Thus, these two kinds of information are integrated for generating a robust map, which can reflect the objects well in the space. Usually 2D map is used for path planning and robot navigation. The Kinect distance information is projected to the ground plane to integrate with that gotten from LRF sensor. Meanwhile, the human detection can be easily proposed by Kinect sensor [37]. From the distance information, the planes in the robot view can be detected by using PCL [38]. After the planes are detected, the ceiling plane can be separated as the highest horizontal plane over 2.5m, and the wall planes can be recognized as the vertical planes that connected to the ceiling plane. For the scene shown in Fig. 2.6 (a), the human detection result is shown in Fig. 2.6 (b). After combining these two kinds of information on the 2D map, the result is shown in Fig. 2.7. The human beings, walls and 3D obstacles like desks are well reflected on the map. A robust map can be generated from the integrated information.

(a) A scene for object detection (b) Detection result by LRF

Figure 2.5: Environment sensing result by LRF.

(a) A scene for object detection (b) Detection result by Kinect

Figure 2.6: Environment sensing result by Kinect.

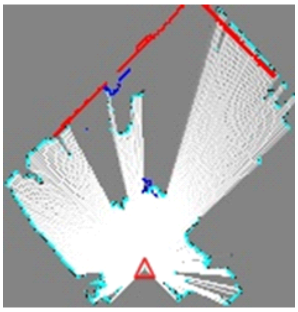


Figure 2.7: Environment sensing result after integrating two kinds of information.

2.2.4 Immobile Area Gird Map Based SLAM

The occupancy grid map used in SLAM is a conventional method which makes a map by an occupancy probability in each grid. When the occupancy probability is high, it means that the probability of existing an object for the grid is high. When the occupancy probability is low, it means that the probability of existing an object for the grid is low, and the probability of noting for the grid is high. The robot keeps sensing the environment around and updating the occupancy probability in each grid. In this way, the areas where objects exist or not can be expressed by probability. However, this method has a problem that even the moving objects would be registered on the map if they are observed in a high probability. As the robust environment map should not contain moving objects and potential moving objects (like a human being keeps still), other methods which can delete the moving objects are used for the researches [39]∼ [42]. To solve the problem, the immobile grid map [35] is applied to generate the map. The environment is divided into girds first, same with the occupancy grid map method. However, the probability for each grid is calculated by the probability of suitable immobile area instead of the occupancy probability. When the occupancy grid map is used, the updating weigh will increase in the exponential way if the object existing probability is high. Moreover, the observed objects might be moving objects under the dynamic environment. The map will be generated correctly only when the objects are still objects as the moving objects should not be shown in the environment map. For the immobile area grid map method, the map updating weight can be adjusted adaptively even if the object existing probability is high, by using a changeable parameter that controlled by the object recognition results.

The event that one grid is immobile area is set as I, and the event that some objects are observed on the same grid is set as O. The immobile area probability $P_t(I)$ can be expressed as Equation (2.1) at time t.

$$P_t(I) \propto P_t(O) \cdot P_{t-1}(I) \tag{2.1}$$

Here, $P_t(O)$ is the object existing probability. From Equation (2.1), the probability $P_t(I)$ that one gird is immobile area at time t can be calculated from the observed object existing probability $P_t(O)$ at time t and the immobile area probability P_{t-1} for the same grid at time $t-1$. The event I can exist depending on the event O can be observed.

Meanwhile, the event that one grid is not immobile area is set as $\overline{I}$, and the event that any objects cannot be observed on the same grid is set as $\overline{O}$. The non-immobile-area probability $P_t(\overline{I})$ can be expressed as Equation (2.2) at time t.

$$P_t(\overline{I}) \propto P_t(\overline{O}) \cdot P_{t-1}(\overline{I}) + P_t(O) \cdot P_{t-1}(\overline{I}) \tag{2.2}$$

From Equation (2.2), the event $\overline{I}$ can exist depending on the event O or the event $\overline{O}$ can be observed. It means the sum of the cases that the event $\overline{O}$ that noting is observed happens and the event O that some objects are observed as moving objects happens. From Equation (2.2), the following equation can be gotten.

$$P_t(\overline{I}) \propto P_{t-1}(\overline{I}) \tag{2.3}$$

From Equation (2.3), the event $\overline{I}$ exists only depending on a constant value. To divide the Equation (2.1) by Equation (2.3), the Equation (2.4) can be gotten.

$$\frac{P_t(I)}{P_t(\overline{I})} \propto \lambda \cdot P_t(O) \cdot \frac{P_{t-1}(I)}{P_{t-1}(\overline{I})} \tag{2.4}$$

Here, λ is the constant coefficient. The updating coefficient for immobile area probability is defined as β, Equation (2.5) can be gotten.

$$\beta = \lambda \cdot P_t(O) \tag{2.5}$$

Here, λ is defined as the variable parameter that is controlled by the object recognition results. The observed objects are recognized as the immobile ares or moving objects first, and the results are used for setting the value of λ. Then we can adjust the value of β by the value of λ. The potential moving objects can be deleted for updating the map by adjusting the value of λ.

The robot senses the environment by its sensors, and the observed objects are recognized first. The value of λ is adjusted depending on the recognition results. If the results are unknown objects, the object is ambiguous and $P_t(O) = 0.5$. The immobile area probability should not be changed as the observed information is meaningless for updating the map. Thus, the value of λ is set as 2 to make sure the updating coefficient for immobile area probability $\beta = 1$. If the recognition results are moving objects, the value of λ is set smaller than 2 so that the observed objects are not used for updating the map. If the recognition results are still objects, the value of λ is set bigger than 2 so that the observed objects are used for updating the map by a big updating coefficient and the objects can be quickly reflected on the map.

The value of λ is set based on the likelihood of the object recognition results. The likelihood of the object recognition results expresses how much the recognition result can be trusted. When the likelihood is low, the objects are better to be thought as unknown objects, and the value of λ is around 2. With the increasing of the likelihood, the value of λ increases from 2 if the objects are recognized as still objects, and the value of λ decreases from 2 if the objects are recognized as moving objects.

2.2.5 Global Mapping of the Work Environment

The map of the work environment is generated in advance. The robot that we used is shown in Fig. 2.8. The Laser Range Finder (LRF) sensor is used for getting the 2D distance information and the Kinect sensor is used for getting the 3D distance information. The height of the LRF sensor is set as 32cm and the Kinect sensor is set as 100cm above the ground on the robot. The mobile platform is chosen as the PIONEER 3-DX made by MobileRobots Co..

Firstly, the experiment is conducted in a room where kinds of objects

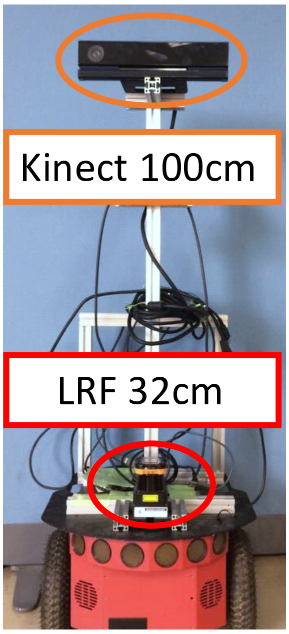


Figure 2.8: The view of the robot.

and a person are inside. The person moved from still status during the experiment. The scene is shown in Fig. 2.9. The generated map by the proposed method and conventional method [11] are shown in Fig. 2.10 and Fig. 2.11 separately. The black points show the immobile areas, the white points show the mobile areas, and the gray points show the unknown areas in the map. By comparing the two results, the chairs and desks are detected as some points in the conventional method as only their legs can be detected by the laser sensor, but they are detected as some areas by the proposed method. These results make sure that the robot cannot go through between the legs as obstacles are there so that collisions can be avoided. The person can be detected while from the beginning even if he was sitting down, and the proposed method did not use human area to update the map. However, the conventional method can only recognize the person after standing up. Moreover, the chair that the person sat before leaving can be also reflected well by the proposed method.

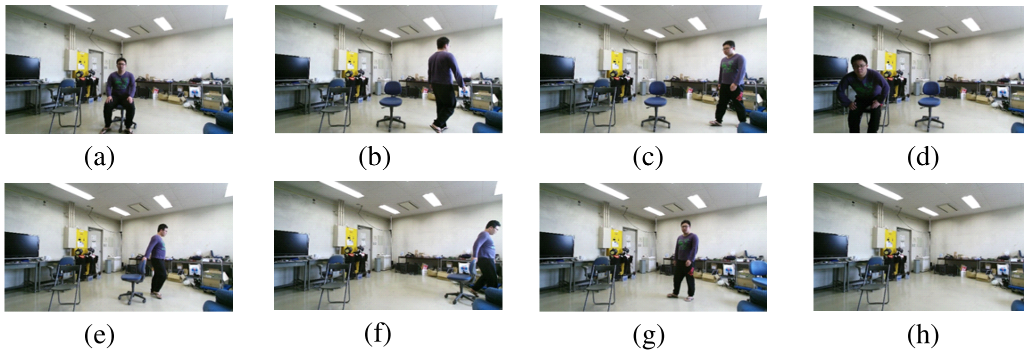


Figure 2.9: The scene of the environment.

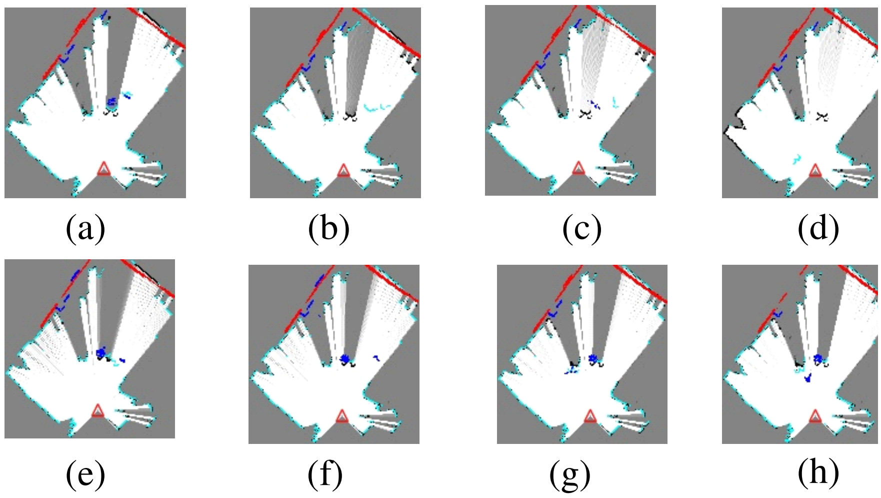


Figure 2.10: Mapping results by the proposed method.

The proposed method is used for generating the map of the lab environment where the guide robot will work. The mapping results are shown in Fig. 2.12. The mobile and immobile areas are reflected well on the generated map.

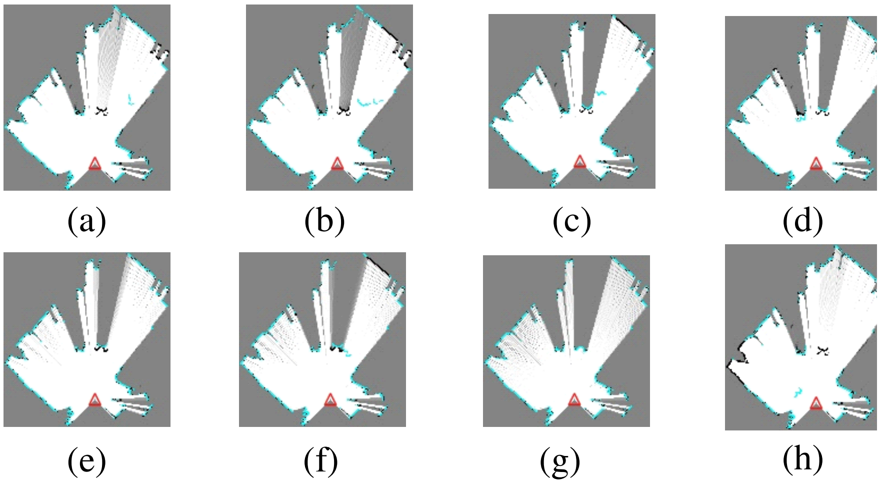


Figure 2.11: Mapping results by the conventional method.

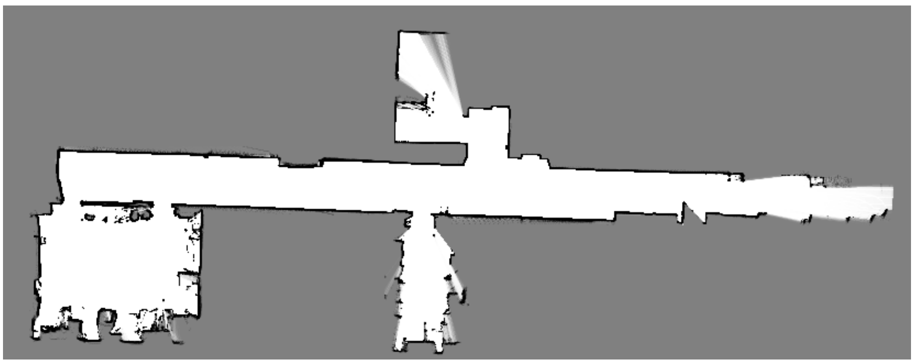


Figure 2.12: Mapping results of the lab environment.

2.3 Path Planning

The method of combining global and local path planning methods is used to generate the path for the robot. The environment model is built in the previous section, and the global map is generated. The Dijkstra's Algorithm based global path planning method is used to calculate the shortest path from the robot to its goal. The subgoals are generated from the shortest path. Then potential field based local path planning method is used to control the robot to move toward the nearest subgoal, which can decrease the possibility to get to local minimum.

2.3.1 Shortest Path Generation

After the environment model is built and the global map is generated, the shortest path can be generated by using Dijkstra's Algorithm. Dijkstra's Algorithm is an algorithm for finding the shortest path between nodes in a graph. After getting the global map, the nodes are set at equal intervals on the global map. In order to make sure the robot can avoid collisions with obstacles in the environment, the obstacle areas in the map are dilated first. The nodes are only generated in the passable areas. That is to say, only the white area of the generated map can be set with nodes. Moreover, the obstacle areas in the map are expanded first before setting the nodes, which can prevent the robot getting too close to the obstacles. This process can make sure the robot move safely by keeping a distance with the obstacles. For the map shown in Fig. 2.13 (a), the nodes are set as 2.13 (b).

When the destination of the robot is set, the Dijkstra's Algorithm is used to calculate the shortest path between the robot and the goal. Firstly, the spaces that are able to pass through are arranged with nodes at equal intervals. Let the node nearest to the starting point be the initial node and the node nearest to the goal be the finishing node. Dijkstra's Algorithm can find the shortest path connecting the starting node and the finishing node.

Under the work environment, the path from the start point to the goal shown in Fig. 2.14 (a) is calculated, and the path planning result is shown in Fig. 2.14 (b).

2.3.2 Path updating under Dynamic Environment

When the robot moves around, it detects obstacles by its own sensors and reflects them to the global map. In this way, the mobile objects which are not registered in the map can be shown in the map, and the passable areas will change with time going on. The shortest path to the goal is calculated and updated on-line. All the time, the robot calculates its shortest path to

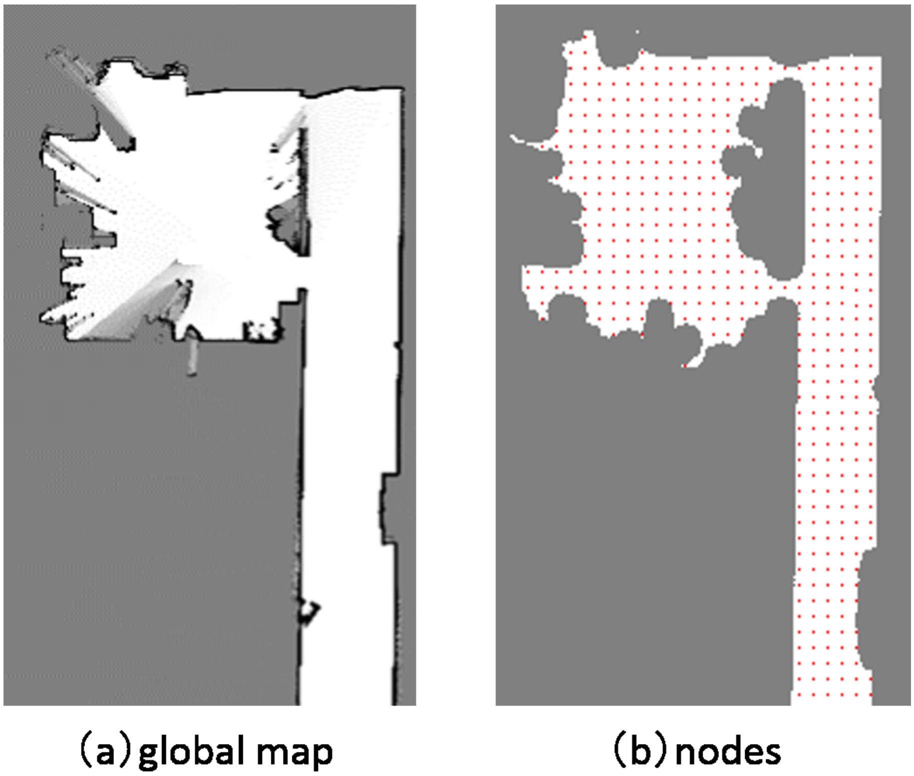


Figure 2.13: Node setting on the global map.

the goal currently, and moves towards the subgoal in the updated path so that the robot can deal with dynamic environment.

2.3.3 Potential Field based Local Path Planning

After getting the shortest path to the goal, the subgoals are generated by sampling from the shortest path at an equal interval. The robot moves toward the current subgoal which is most close to the robot by using potential field method. Potential fields contain attractive potential field and repulsive potential field. The attractive potential field P_a starts from the current subgoal by Equation (2.6):

$$P_a = \frac{k_a}{2} \times d_g^2 \tag{2.6}$$

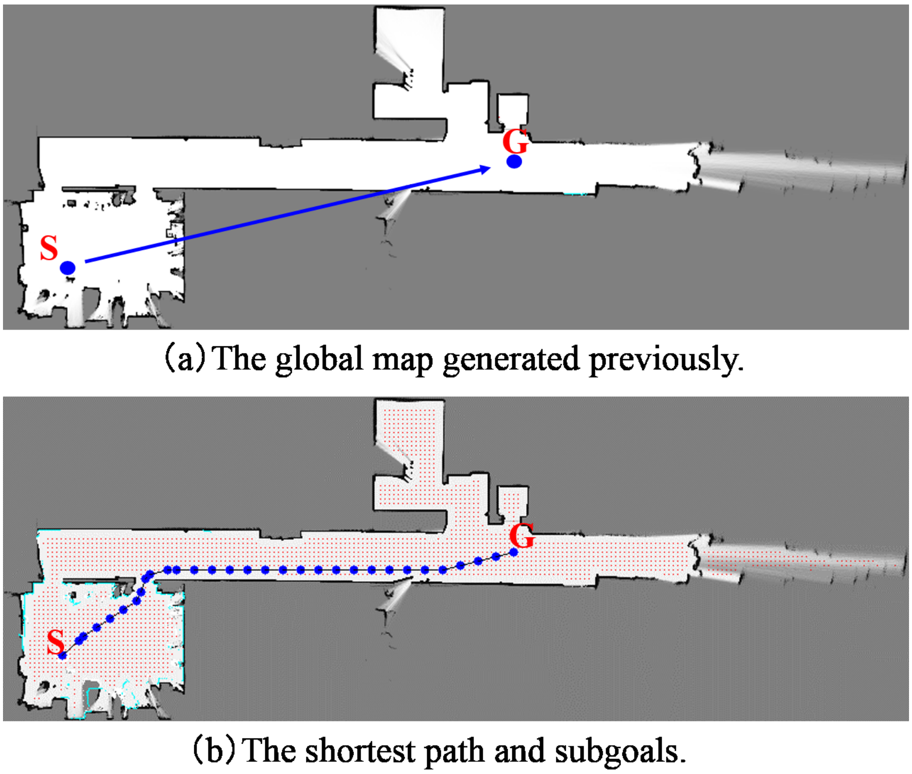


Figure 2.14: The calculated shortest path.

Here, k_a is the attraction coefficient for the goal, and d_g is the distance from the subgoal to the robot.

The repulsive potential field P_r starts from the obstacles by Equation (2.7):

$$P_r = \frac{k_r}{d_r - d_0}, if\ d_r > d_0\ ;\ P_r = \infty, else \tag{2.7}$$

Here, k_r is the repulsion coefficient for the objects, d_r is the distance from the object to the robot, and d_0 is the minimum distance to be traveled by the robot to reach the object.

The shape of the attractive and repulsive potential fields are shown as Fig. 2.15. The integrated potential field can be generated by adding these two kinds of potentials together, and the robot will move toward the place with the lowest potential in the tangential direction.

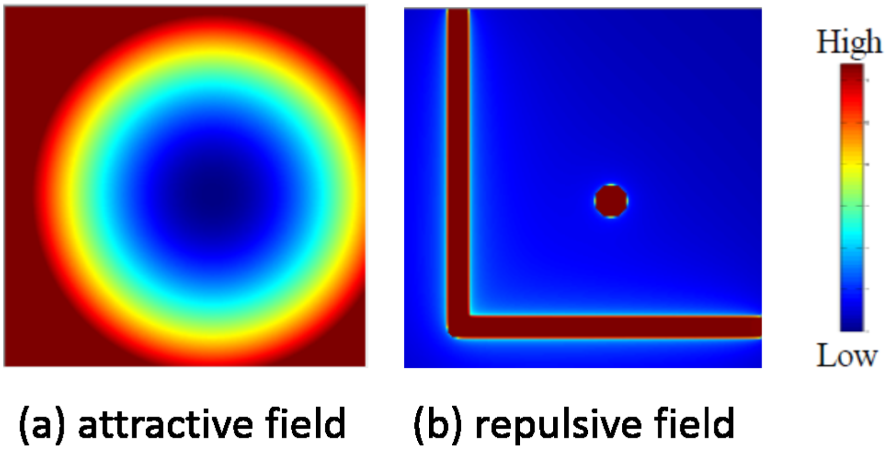


Figure 2.15: Potential fields.

2.4 Conclusion

In this chapter, the immobile grid map based SLAM is applied to generate the 2D global map containing 3D information. The conventional method using laser range sensors cannot deal with 3D obstacles like desks or chairs [35]. By combining the information of 2D and 3D distance sensors, the obstacles in the space are correctly reflected on the map, and potential moving objects like human beings are better recognized and deleted from the map. In this way, a robust global map is generated. Although the global map is generated in advance and can be modified manually, autonomous methods are still needed since the map will be also used for localization when the robot works under the environment. The localization process depends on the matching rate between the scanned information and the global map. Too much modifications manually will also decrease the matching rates. The method proposed in above helps to generate a more robust 2D map containing 3D information without modifications manually, which improves the accuracy of robot self-localization process. Global path planning methods cannot deal with dynamic environment, and local path planning methods cannot make sure the generated path is globally optimum. In this thesis, the path planning process is realized by combining global and local path planning methods.

The shortest path is generated and updated by global path planning method to make sure the users are guided by the globally optimum path. Potential field based local path planning method is used to control the robot to move toward the nearest subgoal while avoiding collisions with obstacles so that the robot can also work under dynamic environment.

Chapter 3

Simultaneous People Recognition and Tracking

3.1 Introduction

People tracking is one of the most researched area in visual surveillance. Detection and tracking people based on camera images has gotten many impressive results [43,44]. Even under some crowded environments [45,46], the users are tracked well with a high accuracy even if the occlusion time is long. However, these methods are usually limited to relatively simple backgrounds or stable illumination conditions. The trajectories of the tracked people in 3D space are also difficult to estimate. Laser range finders, which use lasers to scan the distances from the sensors to the objects, are also often used for people tracking [47,48], and this kind of methods have been successfully applied to public spaces [49] by setting multiple laser sensors. Although these methods work well in practice, they are limited to the estimation of people's positions and to track the people with few occlusions. It is hard to identify the persons while tracking them, especially under crowded environments. When people get cross with others, the tracking system may fail to track the right person, and once it fails, the system cannot restore by itself. The accuracies of laser sensors based tracking methods are also affected by the height of the sensors. For example, the sensors are usually set with the height of over 1m to detect the bodies of adults, but the sensors cannot detect children.

Recently, 3D sensing has been noticed and researched due to the availability of 3D sensors, like Microsoft Kinect sensor. Human detection and tracking methods based on RGB-D information have been proposed [37,50,51]. These works are still difficult to deal with occlusion problems, and the people recognition methods are seldom applied to modify the tracking results. In some recent works [52, 53], researchers have proposed the combination of multiple 3D range sensors and successfully realized people tracking in a public space. However, they need many sensors in their systems, and the methods for calibration are complex.

In this chapter, a simultaneous people recognition and tracking system is proposed. It is a general Markov Chain Monte Carlo (MCMC) particle filter based tracking system by using Kinect sensors. The system is proposed for tracking the particular users for the guide robot. It is also successfully applied to children behavior tracking for the childcare assisting system. In this system, users are recognized first, and the recognition results are used as the observation likelihood for the tracking process. In this way, the users can be tracked robustly. Moreover, the proposed system can modify the errors, which means wrong persons are tracked, by itself once the users are recognized with a high probability.

3.2 Users' Group Recognition and Tracking

During the process that the robot guides users to their destination, the users need to be recognized and tracked, and the tracking results, which shows the status of the users, should be used for controlling the motion of the robot to make it sociable. The overview of the proposed system is shown in Fig. 3.1. The position of each user is detected first. Then the user is recognized by integrating his/her personal information of color and face, and the recognition results are used for calculating the observation likelihood for the tracking process. The tracking problem can be formulated as finding the maximum posterior, and it is solved by using MCMC particle filter. In this way, every user can be recognized and robustly tracked during the guide task. The guide robot that is used for guiding the users is shown in Fig. 3.2. The

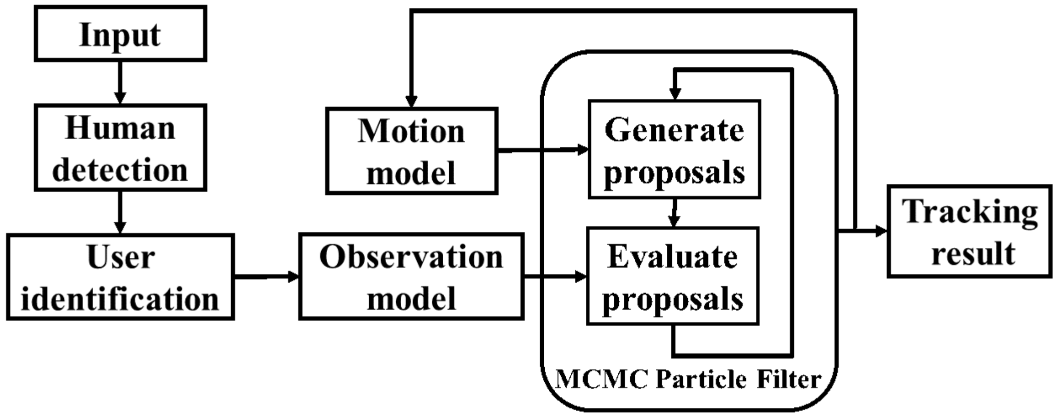


Figure 3.1: Overview of the tracking system.

robot POINEER 3DX is used as our mobile platform. A backward Kinect is set on the robot on the height of 100cm. One of the guide scene is shown in Fig. 3.3. The guide robot moves in front of the users to the destination.

3.2.1 Human Detection

The users are detected by the backward Kinect sensor on the robot. Firstly, the depth information gotten from the Kinect sensor is projected to the ground to generate a XZ map, in which X means the horizontal direction in front of the robot, and Z means the depth direction in front of the robot. For the scene shown in Fig. 3.4, the projected information is shown in Fig. 3.5. The white areas in Fig. 3.5 contain the information of background (circled by pink line), and human areas (circled by red line). The backgrounds are deleted by comparing with the global map that generated in advance. The part of wall lines in the image can be deleted. After that, connected-component labeling [55] process is used to detect out the areas within users' size by Equation (3.1).

$$l_{min} \leq l_{width} \leq l_{max} \ ; \ l_{min} \leq l_{length} \leq l_{max} \tag{3.1}$$

Here, l_{length}, l_{width} mean the width and length of a user candidate area, l_{min}, l_{max} mean the thresholds for a real user area.

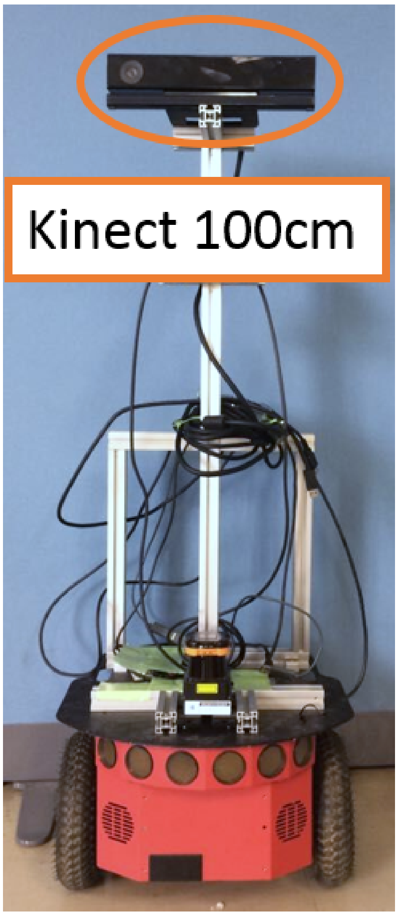


Figure 3.2: View of the robot.

The human areas can be detected in this way, and the human detection results are shown in Fig. 3.6. The five humans are well detected.

3.2.2 Simultaneous Users' Group Recognition and Tracking

The users tracking problem is modeled as a sequential Bayesian framework. A user's state at time t can be expressed as X_t (6 dimensional, location, velocity and acceleration in 2D). When the observation information Z_t is gotten from sensors information at time t, the users states are estimated by

Figure 3.3: Guide scene during the task.

finding the maximum-a-posteriori (MAP) solution of the joint probability. To find the most probable configuration, the MAP solution of $P(X_t|Z_t)$ is estimated by Equation (3.2).

$$P(X_t|Z_t) \propto P(Z_t|X_t) \int P(X_t|X_{t-1}) P(X_{t-1}|Z_{t-1}) dX_{t-1} \tag{3.2}$$

Here, $P(Z_t|X_t)$ represents the observation likelihood at time t, given the sensors input z_t. It measures the confidence of a hypothetical configuration. $P(X_t|X_{t-1})$ is the motion model, which shows the smoothness of the trajectory over time. $P(X_{t-1}|Z_{t-1})$ is the posterior probability of time $t-1$. The posterior probability at arbitrary time t can be calculated from the probabilities from time 1 to $t-1$ sequentially if the posterior probability at initial time is given. The best configuration X_t is then the MAP solution. MCMC

Figure 3.4: Guide scene from the view of the guide robot.

particle filter approximates the MAP solution as a set of discrete samples known as a Markov Chain.

Motion Model

The motion model $P(X_t|X_{t-1})$ can be modeled by giving the update rules as

$$X_t = X_{t-1} + X'_t d_t \; ; \; X'_t = X'_{t-1} + \mu \tag{3.3}$$

Here, μ is a process noise for a user's motion getting from a Gaussian noise.

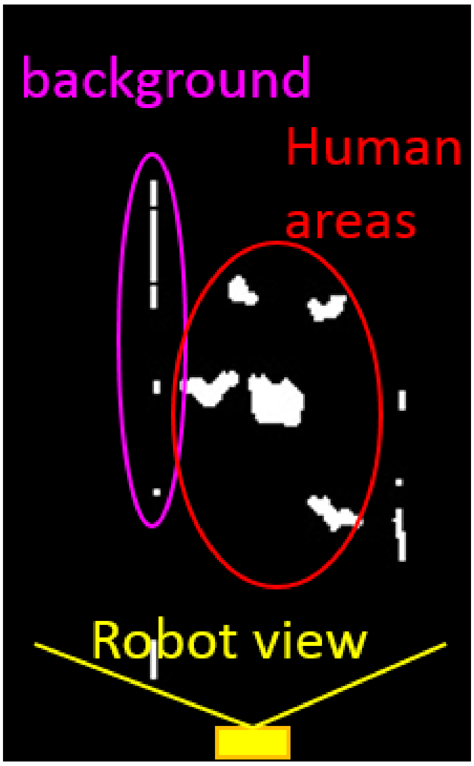


Figure 3.5: XZ image after projection.

Observation Likelihood

Given a hypothesized location of a user on the image, the observation likelihood measures the accuracy of the location. In order to track the users robustly, the particular user recognition results are used as the observation information. In the proposed system, two detectors are used to recognize the particular users: face detector and the color detector. Each single detector has its strength and weakness. The face detector is extremely reliable when frontal faces can be detected, but the face information may not be always available as the users may show their side/back to the sensor. The color detector is almost always available, except the case that the users are totally occluded, but the recognition accuracy is relatively low when different users wear similar clothes or a big part of a users is occluded. The detectors are combined by using a weighted combination of detection responses as shown

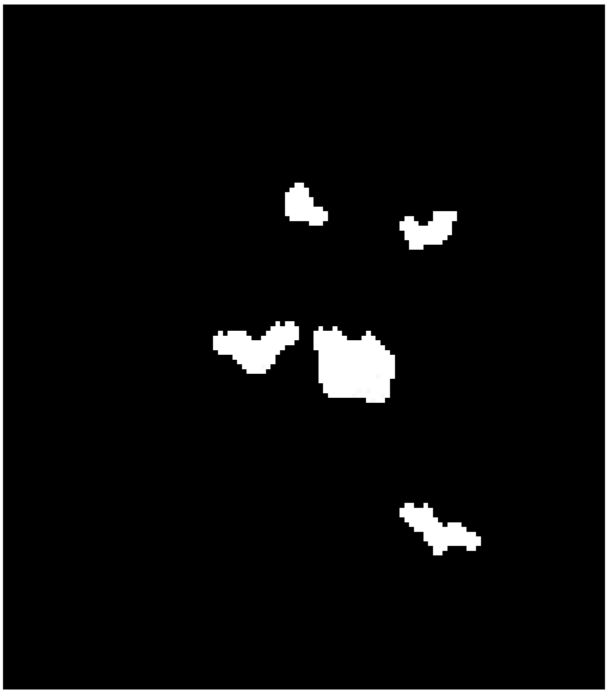


Figure 3.6: XZ image after background subtraction.

in Equation (3.4).

$$P(Z_t|X_t) \propto exp(\sum_j w_j P_j(Z_t|X_t)) \tag{3.4}$$

Here, w_j is the weight of each detector.

Face detector is used to detect and recognize a particular user's frontal face. The OKAOVISION [56] software is used in the system. The particular user face detector likelihood is calculated from the maximum recognition confidence score S_{face}.

$$P_{face}(Z_t|X_t) = \alpha(S_{face} - Th),\ if S_{face} > Th;\ P_{face}(Z_t|X_t) = 0,\ else \tag{3.5}$$

Here, Th is the threshold of face identification confidence. α is the coefficient to adjust the range of the OKAOVISION recognition confidence to 0∼1 so that it can be used as a probability. The weight of face detector is influenced by the showing angle of the frontal face and its distance to the sensor. The angle of the face is changing from $(-\pi/2, \pi/2)$, so w_{face} can be calculated by

$$w_{face} = (1 - 2\theta/\pi)exp(-L),\ if\ FaceDetected;\ w_{face} = 0,\ else \tag{3.6}$$

Color detector is used for searching out the user with similar color. The color information of each user is almost available all the time except the totally occluded cases. In order to decrease the influence of the background, the users are separated from the background. The points in the Kinect point cloud that corresponds to the human detected areas are abstracted and used as the color information of the users. Each user gets a point set that expresses the color of the particular user. This information is matched by histogram matching with the users who need to be tracked. The color histograms of the users are registered in advance. The color images generated from the color point sets are changed into HSV color space first, and the Hue channel and Saturation channel information are used for generating the color histograms. The observation likelihood is calculated from the correlation histogram comparing result S_{color}, which is calculated from Equation (3.7).

$$P_{color}(Z_t|X_t) = S_{color} = \frac{\sum_I (H_1(I) - \bar{H}_1)(H_2(I) - \bar{H}_2)}{\sqrt{\sum_I (H_1(I) - \bar{H}_1)^2 \sum_I (H_1(I) - \bar{H}_1)^2}} \tag{3.7}$$

where

$$\bar{H}_k = \frac{1}{N} \sum_J H_k(J) \tag{3.8}$$

and N is the total number od histogram bins. Here, H_1,H_2 mean the two histograms.

The weight of color detector is designed with the relationship as follows:

$$\sum_j w_j = 1 \tag{3.9}$$

Here, j is face or color.

The observation likelihood is designed like this so that more detectors can be added to the system easily. Two kinds of user recognition detectors are used here so that the observation information contains particular users' recognition information.

Tracking with MCMC Particle Filter

The motion model is used for predicting the status of the users, and the prediction can be evaluated through observation likelihood. Then the MAP solution is found by exploring the space of hypotheses. To effectively explore the configuration space, the MCMC particle filter method is used and extended in three ways:

- A group of independent particles is used to track one user and these particles will be never used for other users' tracking. Multiple users' tracking result can be gotten by parallel running different groups of particles. Different from the conventional MCMC particle filter, the users are not tracked together. Every user is tracked separately so that all of the users can be definitely tracked.

- The conventional estimation result of MCMC particle filter is calculated as the mean value of all the re-sampled particles, but the center of the area that is closest to the mean value of the re-sampled particles is used here as the real position of the user. In this way, the tracking result will be located inside of the detected user area. As the detected user areas are gotten directly from the projection information of the sensor, the tracking results inside the user areas will decrease the errors.

- Multiple users can share one area gotten from the user position detection. When two or more users are close to each other, their projections on the ground may fuse with each other, and their detection result turns out to be a "big" area. Multiple users share this area in practice. In this way, the proposed system can work well even if the detected number of user areas keeps changing. Every user can get a most probable tracking result.

3.3 Experiments

3.3.1 Tracking multiple users under lab environment

In the experiment, the task is set as guiding 5 users from the start point to the goal by leading the users out of a room and passing through the corridor. Four snapshot points are taken from the guiding process to study the tracking results, shown as 4 scenes from Fig. 3.7 to Fig. 3.10. In the beginning, the five users started to move towards the robot. In Fig. 3.7 (b), the yellow rectangle means the guide robot, the five colored circles in the red rectangle means the positions of the five users in the global map. The red rectangle subtitled with tracking results shows the positions of the five users from the view of the guide robot. The blue line connected to the robot means the trajectory. The users' recognition results in this scene are shown in Fig. 3.7 (a). The five users were well recognized in this scene, with four of them were recognized by their faces (small colored rectangles in Fig. 3.7 (a)), and one of them was recognized by color information (big purple rectangle in Fig. 3.7 (a)). All of these five users were tracked well, as shown in the tracking result shown in Fig. 3.7 (b). With time going on, the robot moved out of the room, and the scene 2 shown in Fig 3.8 was observed. In this scene, the five users were also recognized correctly, with four of them were recognized by their faces (small colored rectangles in Fig. 3.8 (a)), and one of them was recognized by color information (big light blue rectangle in Fig. 3.8 (a)). The user who was recognized by color is changed. The robot moved smoothly out of the room (blue trajectory shown in Fig. 3.8 (b)), and all of the users were still well tracked, as shown in the tracking result of Fig. 3.8 (b). Then the robot stated to turn its moving direction and the users started to get out of the view of the robot. Fig. 9 shows the users' recognition results and their tracking results at this time. Three of the users were recognized by their faces (small colored rectangles in Fig. 3.9 (a)), and the other two users were not detected actually. However, the system was designed as that all of the five users had to be recognized from the detected users anyway even if the number of the detected people was less than the real number. Thus the left two users who could not be recognized by the face information were recognized as the detected people with similar colors. As the results, the other two users who were recognized by color are shown in Fig. 3.9 (a) as the big purple and green rectangles. Although these two users were not recognized correctly, they were still well tracked, since the tracking process smoothed the trajectories of the users. The tracking results are shown as the red rectangle subtitled with tracking result in Fig. 3.9 (b). After that, the robot moved on, and the users were well tracked, as shown in Fig. 3.10. The robot moved through the corridor and the users followed it. In this scene, the five users were again recognized correctly, with four of them were recognized by their faces (small colored rectangles in Fig. 3.10 (a)), and one of them was recognized by color information (big purple rectangle in Fig. 3.10 (a)). The tracking results observed from the view of the robot is shown as the red rectangle subtitled with tracking result shown in Fig. 3.10 (b).

The five users are tracked correctly during the guiding process. The relative positions of the users with the robot can be calculated. In this way, the robot can understand the status of the users and adapt to the motions of the users. The adaptive motion framework is explained in the next chapter.

3.3.2 Comparison with conventional image based RJ-MCMC particle filter tracking method

To show the effectiveness of the proposed method, the same robot is used to track multiple users with the conventional image based Reversible Jump Markov Chain Monte Carlo (RJ-MCMC) particle filter method. To decrease the miss detections of humans, the background subtraction is usually used. Subtraction Stereo is also used to detect the moving object. However, these methods may not work for a mobile robot since the background keeps changing if the robot moves, or the humans may keep static. Thus, the background area in the robot's view is detected in real time, and the human detection process is restricted to the foreground area. For the indoor working environment, the walls, floor and ceiling are considered to be background. The backgrounds are detected by using plane detection. The planes are con-

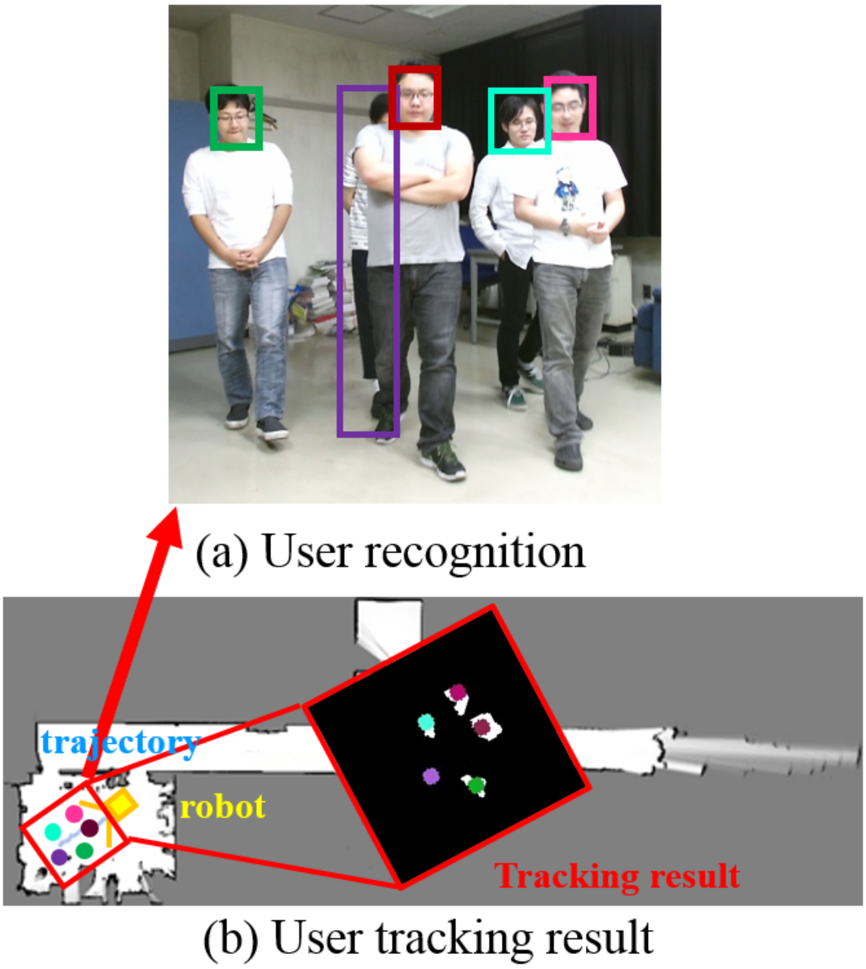


Figure 3.7: User detection and tracking results (scene 1).

sidered to have the same normal direction, and the parallel planes can be divided by using the distance information. The shallow problem can also be solved as the floor plane is deleted before detection process. The humans are detected by using joint-HOG features and feet ellipse fitting. The particular users group is distinguished by using Local Binary Pattern (LBP) histogram matching based face recognition. Notice that the face information of the users to be guided is automatically initialized at the beginning. The face areas are detected by using Haar-like features, and only when the face re-

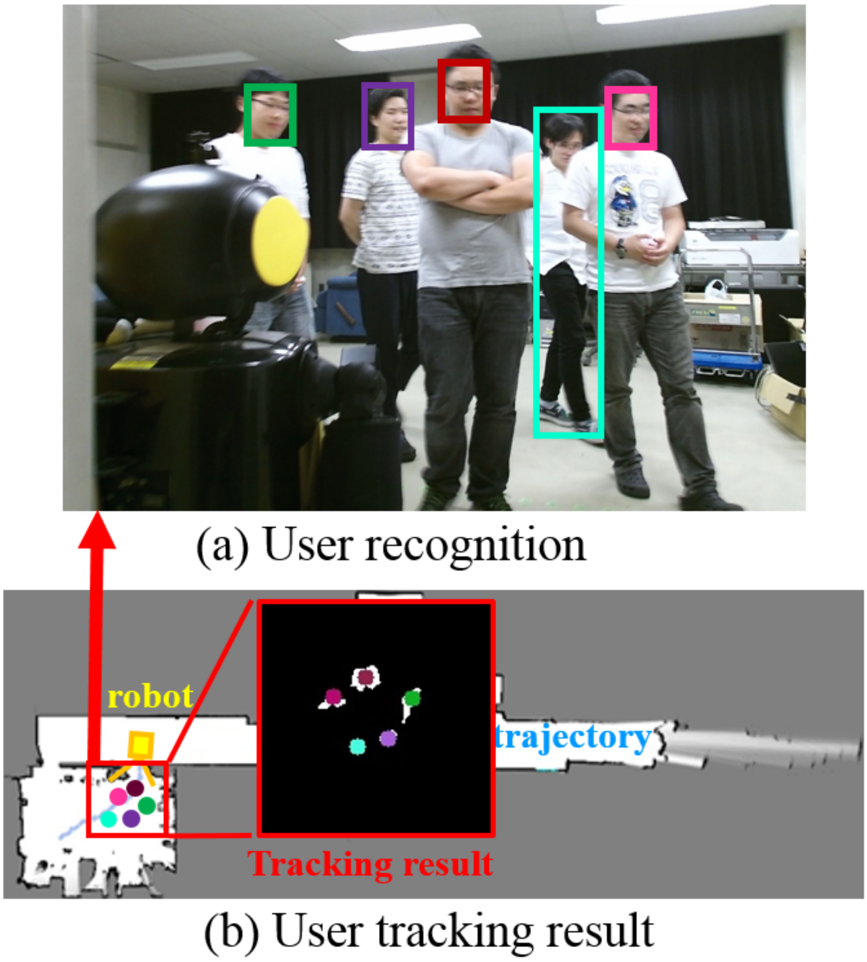


Figure 3.8: User detection and tracking results (scene 2).

gions are detected, the face recognition process works. The tracking work is realized by using RJ-MCMC based particle filter. The users can be tracked on the image and the positions of the users and the robot can be estimated by matching the immobile feature points in the robot view. The detail of the conventional methods are explained in [77,78]. From the results, the users are also well tracked on the image in the conventional method, but the accuracy of position information is far from enough to show their motion behaviors. To know the trajectories of the users, the proposed method is better.

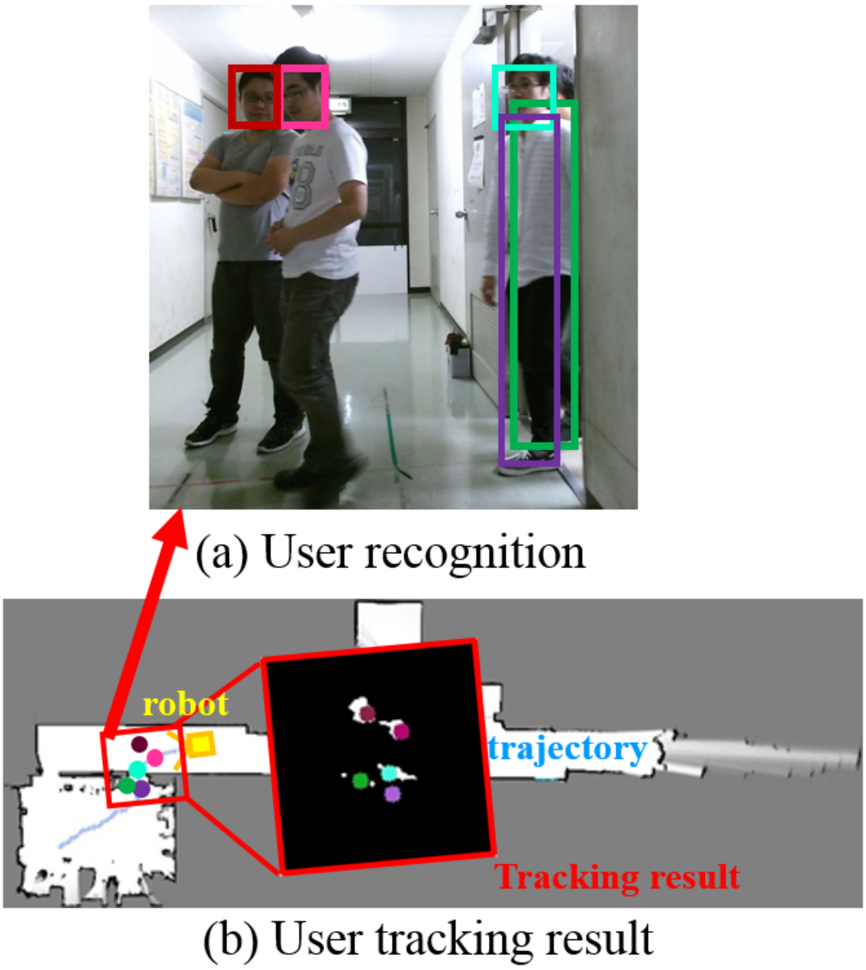


Figure 3.9: User detection and tracking results (scene 3).

3.4 Application to the childcare assisting system

In this chapter, a simultaneous people recognition and tracking system is proposed. It is a general Markov Chain Monte Carlo (MCMC) particle filter based tracking system by using Kinect sensors. The system is proposed for tracking the particular users for the guide robot. It is also successfully

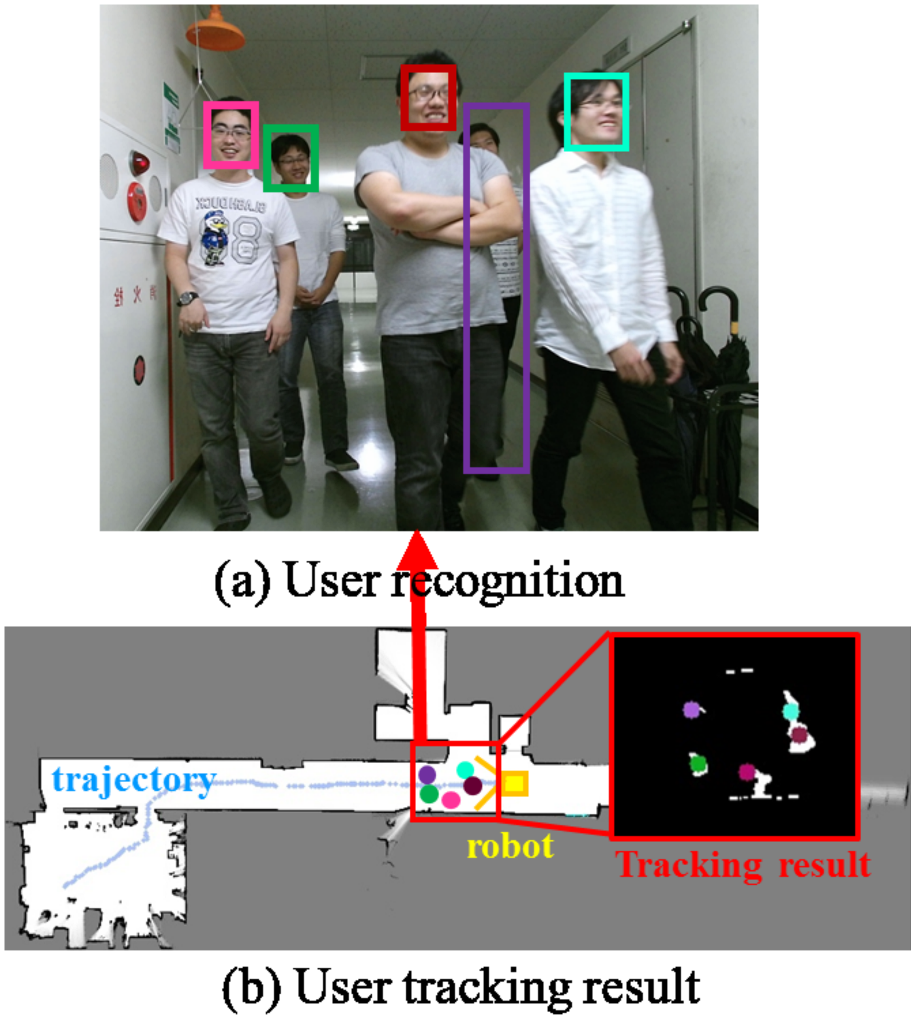


Figure 3.10: User detection and tracking results (scene 4).

applied to children behavior tracking for the childcare assisting system. The data of a group of children playing games in a classroom is used to test the usefulness of the proposed system, which contains more difficult scenes than the adult people group as the motions of the children are more unpredictable. The proposed system is proved effective by robustly tracking the children behaviors in a nursery school.

The children data are gotten from the research project of developing a childcare assisting system. In recent years, double-income households keep increasing, and more and more people want to send their children into nursery schools. Low birthrates have become a critical problem all over the world as some developed countries are rapidly becoming extremely old societies. As a social problem, it is the duty of the governments to improve the school environment for the children. Parents also hope their children could be raised in a qualified nursery schools. However, the number of qualified nursery teachers is far from enough [57]. Childcare assisting system can be used as one way to assist the nursery teachers with their work. There are several researches about developing childcare assisting systems for nursery schools [58] [59]. These researches makes some contributions on understanding the behaviors of the children. However, the sensor networks or wearable sensors that they used are difficult to set and require the cooperation of the children. The nursery teachers are very interested in childcare robotic systems because it would be helpful to monitor children's activities and to play with the children [60]. Developing a childcare assisting system, supporting functions of which are designed from the viewpoints of the nursery teachers, is greatly needed. This childcare assisting system is supposed to be applied to nursery schools in the near future to support the work of nursery teachers. The functions of the childcare assisting system is designed based on the investigations of the nursery teachers' requirements. As a basic function, the childcare assisting system should be able to track the behaviors of children in natural states so that it provides useful information to the nursery teachers.

In this section, the proposed tracking system is applied to the robust children behavior tracking system by using multiple Kinect sensors. To solve the occlusion problem, multiple Kinect sensors are integrated, which are set from different views in different height. The children are detected and recognized by integrating his/her personal features of face and color. The tracking process is realized by using Markov Chain Monte Carlo (MCMC) particle filter method. The number of Kinect sensors is adjustable according to the size of the space. The experiments were conducted in a childcare school, as shown in Fig. 3.11.

Figure 3.11: The tracking scenes in a nursery school.

3.4.1 Children Detection

In order to track the behaviors of the children in a nursery school, multiple Kinect sensors were set in the classrooms. The number of Kinect sensors can be adjusted according to the space of the classroom. Here, an example of using two Kinect sensors for a real classroom of nursery shown in Fig. 3.11 will be explained in detail as an example, and more sensors can be added to the system in the same way if the views of the two Kinect sensors are not enough. The sensor positions in the classroom are shown in Fig. 3.12. Kinect 1 is set in the average height of children in front of the class to monitor the children with high qualified frontal face images, and Kinect 2 is set slanted in a higher height to monitor all of the children with less occlusions. The accurate positions and slanted angles are not required in the sensor setting step, which ensures that the sensing equipment is easy to mount. The accurate coordinates of Kinect sensors will be estimated during the calibration process. The scene shown in Fig. 3.11 is taken from the slanted Kinect 2.

Firstly, the positions of the children are detected by the two sensors separately. Here, the slanted Kinect 2 is taken as an example to explain the

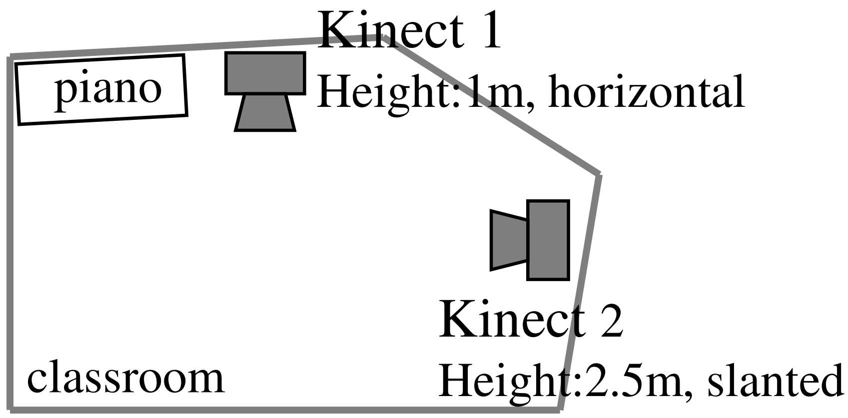


Figure 3.12: Sensors placement in the classroom.

detection algorithm. As the sensors are set roughly, we firstly estimated the accurate height and slanted angle of the Kinect 2. All the planes are detected out using Point Cloud Library (PCL) [38] in the empty classroom, and segmented out the sub-horizontal plane with the lowest height as the floor plane. From the slanted angle of the floor plane, the accurate Kinect slanted angle can be calculated. This process needs to process only once as initialization after setting the sensors. The children are detected by projecting the transformed point cloud with the child height range (0.5m-1.2m) on the ground, and deleting the background parts. Similarly, labeling process is used to detect out the areas within children size by Inequality Equation (3.10).

$$l'_{min} \leq l'_{width} \leq l'_{max} \; ; \; l'_{min} \leq l'_{length} \leq l'_{max} \tag{3.10}$$

Here, l'_{length}, l'_{width} mean the width and length of a child candidate area, l'_{min}, l'_{max} mean the thresholds for a real child area.

The detection process is shown in Fig. 3.13. For the scene shown in Fig. 3.11, the projecting result on the ground plane is shown in Fig. 3.13 (a). The white areas contain the children position information and backgrounds. The final children position detection result is shown in Fig. 3.13 (b). The backgrounds areas are deleted from the projecting result. The detection results were compared with the correct ones that we made manually from the point cloud one frame by one frame, and found that the average error for each child is around 10cm, with the standard deviation 8cm. The detection results showed the position coordinates and their relative position relationships correctly. In this way, the children position information can be gotten from the two Kinect sensors.

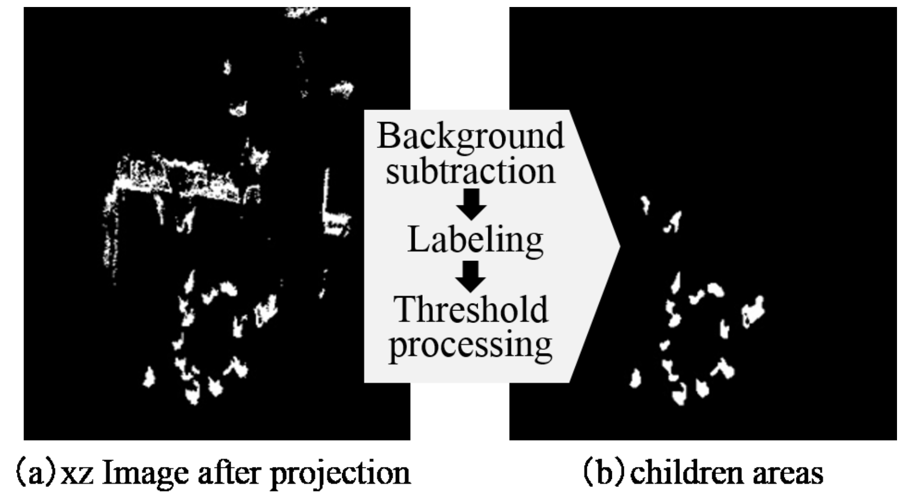


Figure 3.13: Children detection process.

3.4.2 Calibration of Multiple Kinect Sensors

The 3D reconstruction resulting from multiple sensors strongly depends on a good calibration result. The calibration problem is solved by matching the same corresponding points of two Kinect sensors. The matching is processed by matching the corresponding points on 2D position maps to decrease the complexity and calculation. The corresponding points are matched by affine transformation. The affine transformation matrix is calculated in advance during the initialization process. A series of corresponding points in 2 Kinect coordinate systems can be gotten easily by asking a person walking around in the classroom after setting the sensors. The single detected person by the two Kinect coordinate systems are surely be the same person. 3 of the corresponding points are chosen to calculate the affine transformation matrix. The remaining points are used to check the residual sum of squares (RSS). This process is repeated to find the best affine transformation matrix with the least RSS. Finally, the best affine transformation matrix for calibration is chosen out. Notice that the calibration only needs to be processed once after setting the sensors during the initialization process.

3.4.3 Simultaneous Children Recognition and Tracking

The children tracking problem is modeled by using a sequential Bayesian framework. A child's state at time t can be expressed as X_t (6 dimensional, location, velocity and acceleration in 2D). When the observation information Z_t is gotten from sensors information at time t, the children states are estimated by finding the maximum-a-posteriori (MAP) solution of the joint probability. To find the most probable configuration, the MAP solution of $P(X_t|Z_t)$ is estimated by Equation (3.11).

$$P(X_t|Z_t) \propto P(Z_t|X_t) \int P(X_t|X_{t-1}) P(X_{t-1}|Z_{t-1}) dX_{t-1} \tag{3.11}$$

Here, $P(Z_t|X_t)$ represents the observation likelihood at time t, given the sensors input z_t. It measures the confidence of a hypothetical configuration. $P(X_t|X_{t-1})$ is the motion model, which shows the smoothness of the trajectory over time. $P(X_{t-1}|Z_{t-1})$ is the posterior probability of time $t-1$. The posterior probability at arbitrary time t can be calculated from the probabilities from time 1 to $t-1$ sequentially if the posterior probability at initial time is given. The best configuration X_t is then the MAP solution. MCMC particle filter approximates the MAP solution as a set of discrete samples known as a Markov Chain.

Motion Model

The motion model $P(X_t|X_{t-1})$ can be modeled by giving the update rules as

$$X_t = X_{t-1} + X'_t d_t \ ; \ X'_t = X'_{t-1} + X''_t d_t \ ; \ X''_t = X''_{t-1} + \mu \tag{3.12}$$

Here, μ is a process noise for a user's motion getting from a Gaussian noise.

Observation Likelihood

Given a hypothesized location of a child on the image, the observation likelihood measures the accuracy of the location. In order to track the children robustly, the particular child recognition results are used as the observation information. In the proposed system, three detectors are used to recognize the particular children: a face detector, a color detector and a motion detector. The motion detector is added since it can effectively limit the motion range of the child as he/she cannot move a long distance in a single frame time. The detectors are combined by using a weighted combination of detection responses as shown in Equation (3.13).

$$P(Z_t|X_t) \propto exp(\sum_j w_j P_j(Z_t|X_t)) \tag{3.13}$$

Here, w_j is the weight of each detector.

Face detector is used to detect and recognize a particular child's frontal face. The OKAOVISION [56] software is used in the system. The particular child face detector likelihood is calculated from the maximum recognition confidence score S_{face}.

$$P_{face}(Z_t|X_t) = \alpha(S_{face} - Th), \ if S_{face} > Th; \ P_{face}(Z_t|X_t) = 0, \ else \tag{3.14}$$

Here, Th is the threshold of face identification confidence. α is the coefficient to adjust the range of the OKAOVISION recognition confidence to 0～1 so that it can be used as a probability. The weight of face detector is influenced by the showing angle of the frontal face and its distance to the sensor. The angle of the face is changing from $(-\pi/2, \pi/2)$, so w_{face} can be calculated by

$$w_{face} = (1 - 2\theta/\pi)exp(-L), \ if \ FaceDetected; \ w_{face} = 0, \ else \tag{3.15}$$

Motion detector is a strong indicator of the presence of a person. The area around the predicted position usually has a higher possibility to be the tracked target. The observation likelihood is calculated from the distance D between the predicted position and detected children areas.

$$P_{motion}(Z_t|X_t) = \beta/D^2 \tag{3.16}$$

Here, β is the coefficient to adjust the range of the motion likelihood.

Color detector is used for searching out the user with similar color. The color information of each user is almost available all the time except the totally occluded cases. In order to decrease the influence of the background, the users are separated from the background. The points in the Kinect point cloud that corresponds to the human detected areas are abstracted and used as the color information of the users. Each user gets a point set that expresses the color of the particular user. This information is matched by histogram matching with the users who need to be tracked. The color histograms of the users are registered in advance. The color images generated from the color point sets are changed into HSV color space first, and the Hue channel and Saturation channel information are used for generating the color histograms. The observation likelihood is calculated from the correlation histogram comparing result S_{color}, which is calculated from Equation (3.17).

$$P_{color}(Z_t|X_t) = S_{color} = \frac{\sum_I (H_1(I) - \bar{H}_1)(H_2(I) - \bar{H}_2)}{\sqrt{\sum_I (H_1(I) - \bar{H}_1)^2 \sum_I (H_1(I) - \bar{H}_1)^2}} \tag{3.17}$$

where

$$\bar{H}_k = \frac{1}{N} \sum_J H_k(J) \tag{3.18}$$

and N is the total number od histogram bins. Here, H_1,H_2 mean the two histograms.

The weights of motion and color detector are designed with the relationship as follows:

$$w_{motion} = \lambda w_{color}; w_{face} + w_{color} + w_{motion} = 1 \tag{3.19}$$

Here, λ is a constant value, showing that the weight of motion is λ times of the weight of color.

The observation likelihood is designed like this so that more detectors can be added to the system easily. Two kinds of user recognition detectors are used here so that the observation information contains particular users' recognition information.

Tracking with MCMC Particle Filter

The motion model is used for predicting the status of the users, and the prediction can be evaluated through observation likelihood. Then the MAP solution is found by exploring the space of hypotheses. The extended MCMC particle filter is applied to the system.

3.4.4 Children Recognition and Tracking

It is a challenge to track the children when they move naturally. The difficult scene of the drum game during a eurythmic class is chosen to show the usefulness of the proposed system. The teacher leaded the children to walk or run along with drum rhythm. Figure 3.14 shows the tracking results of the teacher and two particular children. For each frame, 5 pictures are shown to express the status of the children: " (a) color " shows the color image of the current frame and the persons that to be tracked; " (b) depth " shows the position information of the detected teachers and children. The Kinect 2 is set in the middle-bottom position in the picture; " (c) teacher " shows the trajectory of the teacher during the game; (d) and (e) show the trajectories of two children during the game. The teacher leaded the children walk in the clockwise direction for two circles. The tracking results show the motion tendency and trajectories of all the people perfectly. In frame 120, it is observed in (b) that the child 2 and the teacher share one same area in the depth image as they are close to each other. It solves the problem that less people are detected than the real number. This may leads to some errors of the position of the tracked child as the output will be the center of all the long area. However, this kind of error do not affect the motion tendency so much of the tracked children so that the trajectories can still show the motion of the children robustly. During the whole game, the teacher leaded the children move in the clockwise direction for four circles. Their moving speed kept changing during the game. All the trajectories responded the motion tendency of the tracked persons well. In order to show the validity of

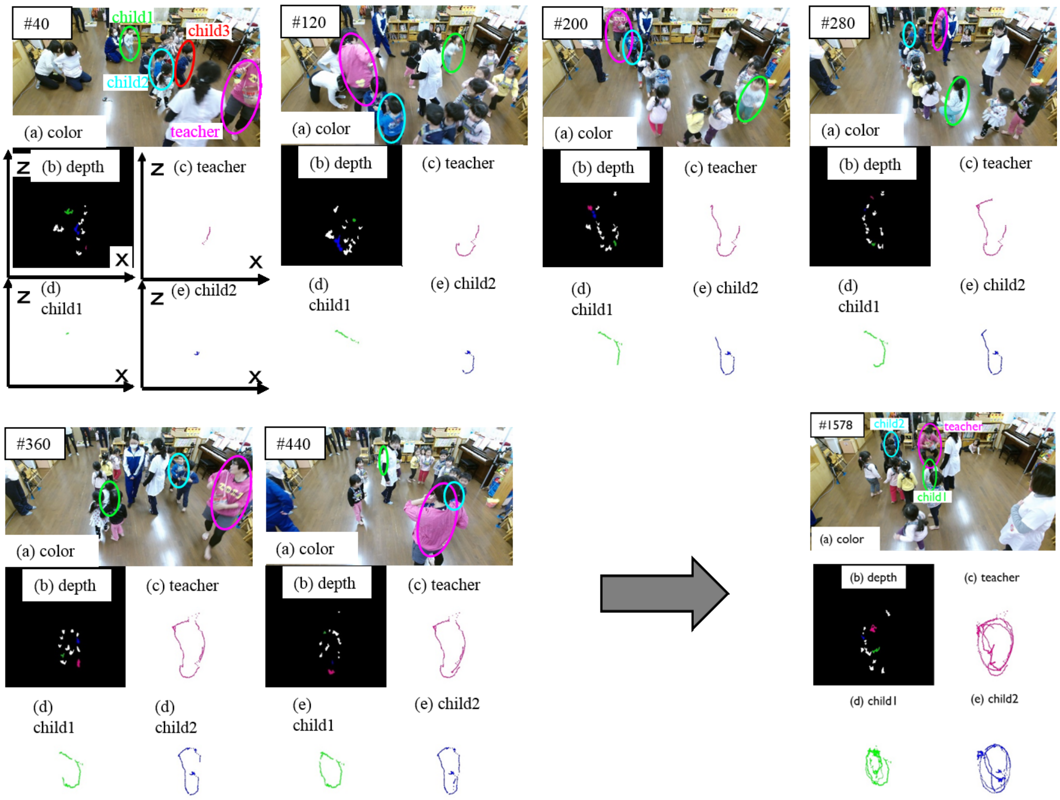


Figure 3.14: The tracking trajectories of the teacher and two children during the drum game.

the proposed system, the tracking results are evaluated by comparing them with the correct ones. The correct results are generated manually by assigning the position from original information of Kinect sensors. The tracking accuracies for each child can be calculated. Figure 3.15 shows the tracking error changing tendency of the teacher and the child 1. The average errors of the tracking result of them are 0.103m and 0.122m, with the standard divisions as 0.088m and 0.112m. This tracking is good enough for our purpose of analyzing the behavior of the children. The teacher and the child 1 are successfully tracked by the proposed system during the whole drum game, although the distance errors becomes huge for a few frames. To show the effectiveness of the system, the tracking results are also compared with that generated from the conventional multiple laser sensors based tracking method [49]. The tracking results by the conventional method are shown in Fig. 3.16. It is observed that the conventional method can not track the people well even if it can detect out their positions. For example, the teacher can be only tracked for 46s during the drum game (80s in total). After that (frame 948 in Fig. 3.16), the teacher is also detected as a person, but recognized as another one, which leaded to the failure of continuous tracking (frame 948 ∼ frame 1578).

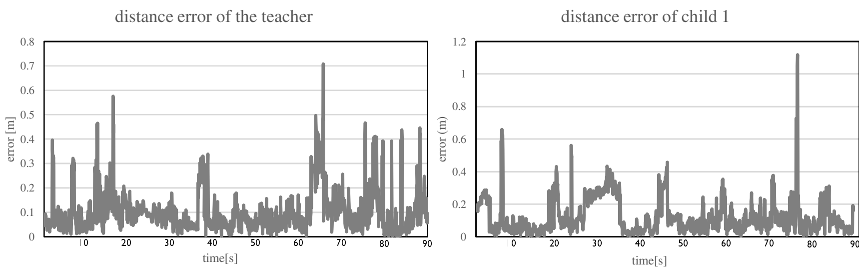


Figure 3.15: The tracking error changing tendency of the teacher and the child 1.

To get the growth information of the children, they need to be continuously observed for a long term. Up to now, the children have be continuously observed every three moths for more than one year. Another scene that the children are tracked in a different class is shown as Fig. 3.17 (a). It shows that the children were playing the same game with the teacher. The human detection results are shown in Fig. 3.17 (b). The white areas shows the positions of all the people in the scene. In the children recognition process, the face and color information for each person is recorded during the initialization process. Here 5 frontal face photos are registered for each person, and the color histogram of Hue and Saturation channels in HSV color space is used as the color information. Figure 3.18 shows some face detection and identification results during the class activities. The orange rectangles in the picture show the face recognition results. Most of the children are recognized with a likelihood. For the children who cannot be recognized by face, they

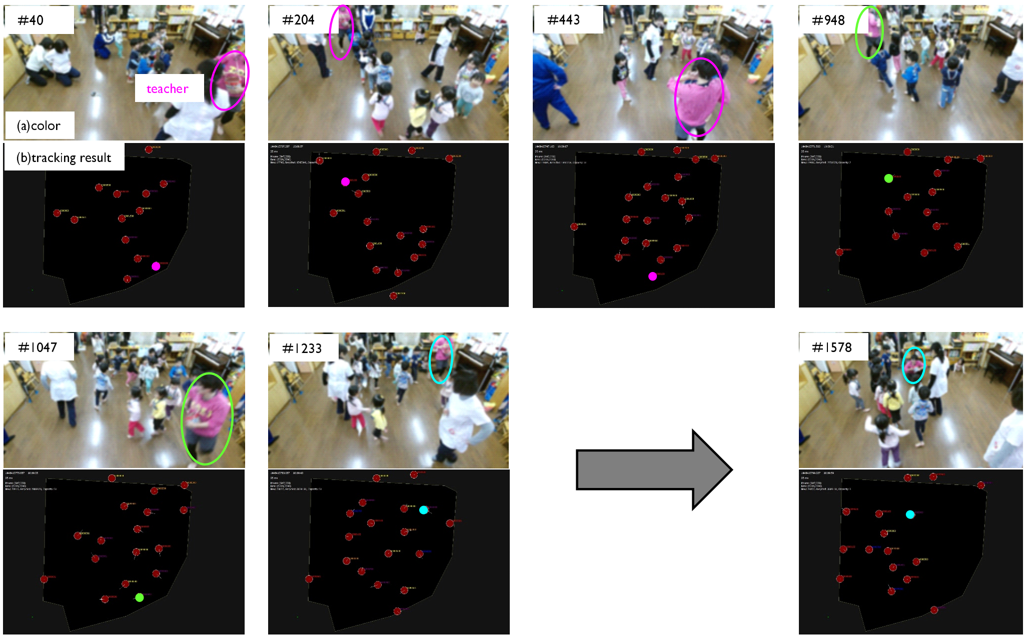


Figure 3.16: The tracking result based on multiple laser sensors during the drum game.

are recognized by color information (circled by green line).

(a) the scene during a drum game (b) the detected human areas

Figure 3.17: Children detection process.

The tracking result of a particular child during the drum game is shown in Fig. 3.19. From the color image, it is observed that during frame 0-100, the face of the particular child (U1) was well detected and identified as the child

(a) the scene of sitting together (b) the scene during the drum game

Figure 3.18: Face based identification and color based identification.

almost stayed in the same position. After that, U1 stated to run with other children and his face was unable to be detected. During this process, color information was used for recognizing the child in the color image. When his face was identified again, the system relied on the face identification result to track his face. From the detection result, it can be also observed that the children and the teachers are well detected. However, their areas may fused with each other when they were close with each other. The proposed tracking method is robust to track U1 during the game. Even under some special conditions like frame 180, U1 was close with another child and detected as a big area in the ground plane ((b) detection of frame 180 in Fig. 3.19). Our system treat this area as U1 and this result could be thought as correctly tracked. Actually this area was also identified as other child in the same frame, which proved the effectiveness of the improvement on MCMC particle filter. The motion trajectory is shown in Fig. 3.19 (d). In order to show the validity of the system, the tracking results is also evaluated by comparing with the correct ones. The correct results are generated manually by assigning the position from original information of Kinect sensors. Figure 3.20 shows the tracking error changing tendency of the child. The average error of the tracking result is 0.193m, with the standard divisions as 0.122m. This tracking accuracy is good enough for the purpose of tracking and understanding the behavior of the children and providing necessary information to the nursery teachers. U1 is successfully tracked by the system during the whole drum game with low error, although the distance errors becomes a

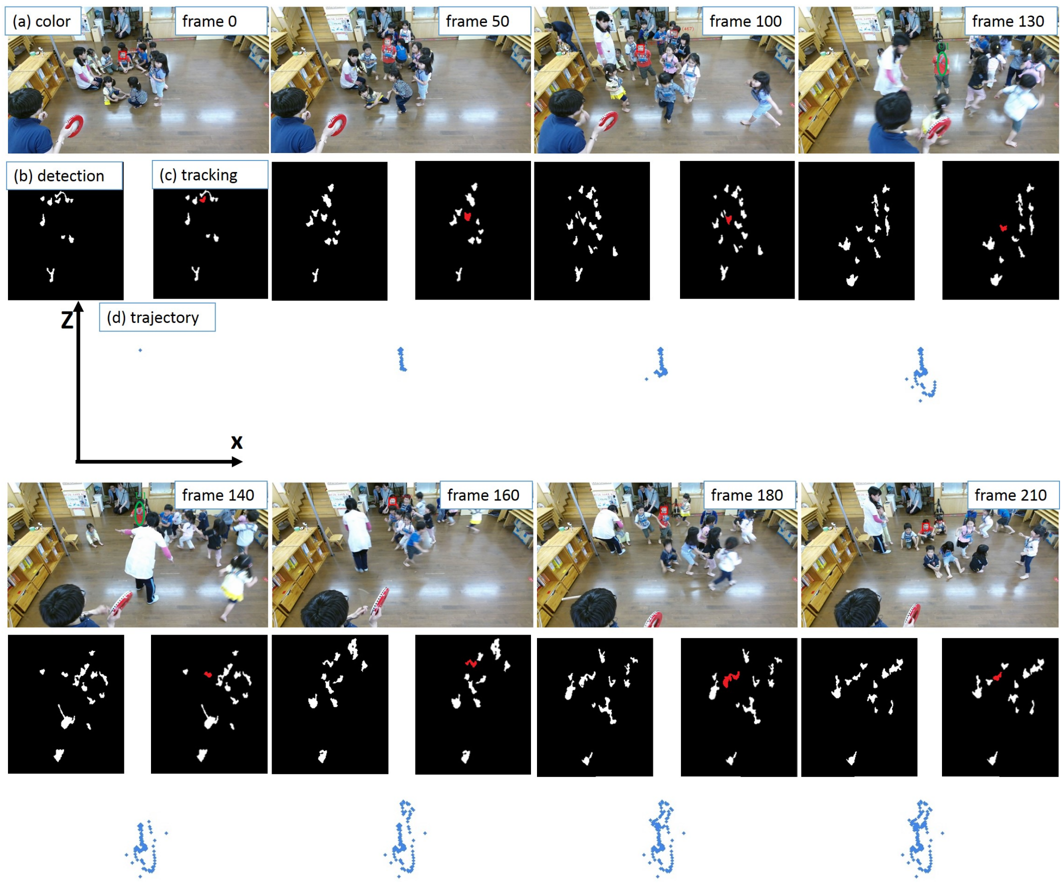


Figure 3.19: The tracking trajectories of a child during the drum game.

little big for a very short time. These errors are caused by the overlapping of projected positions on the ground. As more than one child are closed to each other, they formed a “ big ” area in the human detection result. The center of this area is different from the real position of any single child. However, the proposed method is more robust as the error would decrease after these children are separated with each other, which is shown as breaking up of the “ big ” area in the detection result.

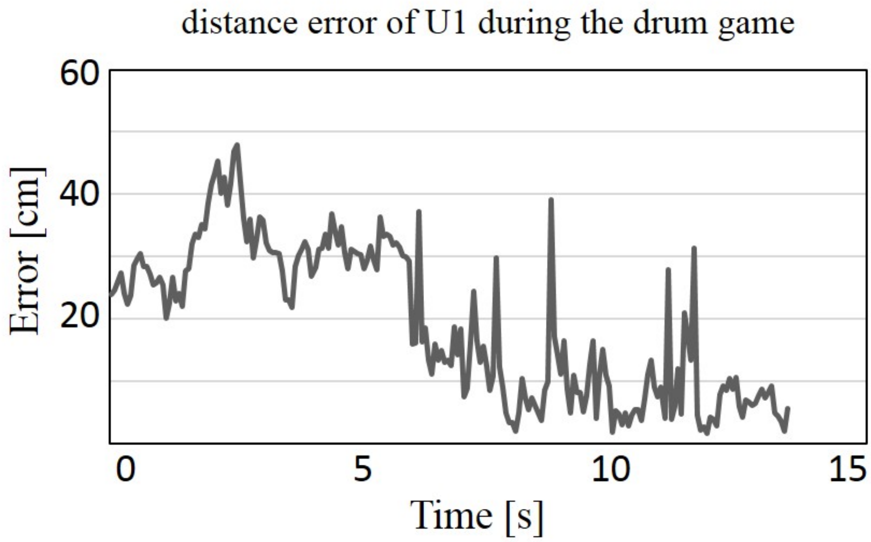


Figure 3.20: Tracking error of the particular child during the drum game.

3.4.5 Children Behavioral Analysis

In order to provide useful information for the nursery teachers to help their work, the children behaviors are also analyzed based on the requirements of the nursery teachers. The nursery teachers believe that the motion range and momentum of a child can show the growth process. A younger child or a new member tends to be quiet. They will become more active with growing up. After long term of observation, the growth of the children can be evaluated by quantitative methods. Meanwhile, the relationship among the children and the dependence to the teacher can also show the growth of the children. The teachers need to adjust team work games during the class to make sure every child grows up healthy in psychology.

Motion trajectories of the children

As the nursery teachers need to record the activities of the children after class, they used to have to remember all the reactions or motions during the whole class. This is almost impossible since the amount of information is too huge. They can only remember some special reactions of a child and the performances of some special (very active or uncooperative) children. The proposed system can provide the motion trajectory of any child. This information can help the nursery teachers remember the performance of any child. Figure 3.14 have shown the motion trajectories of some children.

Motion Range

The nursery teachers believe that the motion range and momentum of a child during the class can show the growth process and familiarity to the class. A younger child or a new member trends to be quiet. They will become more active with growing up. Motion range can be used as a quantitative index to show the growth of a child. The childcare robot also needs to know this information to decide its motion range for best adapting to the children. The proposed system can provide accurate motion range information of the children. From the tracking results, the motion area of a child can be calculated by finding the bounding rectangle of his/her trajectory. Their dynamical momentum can also be calculated from the length of the trajectories. The motion areas of the teacher and the two children are shown in Fig. 3.21 (a) and their dynamical momentums are shown in Fig. 3.21 (b). With these information, the system can be applied to monitor the developments of a child along with the growth of their ages after a long term observation.

Relative distance

The nursery teachers need to know the relationship among the children for better leading their growth. Besides, the teachers also need to know how much a child relies on him/her in the daily life. This can be evaluated by the relative distances between two persons. From the tracking results, the relative distance between different persons can be calculated accurately. Fig. 3.22 (a) shows the relative distances between the teacher and 3 different children. Their average relative distances during the game is 1.768m, 0.797m, 0.550m with the standard deviation of 0.689m, 0.416m, and 0.267m. It can be observed that child 2 and child 3 prefer to stay close to the teacher, and child 1 prefers to keep a small distance with the teacher. Similarly, Fig. 3.22

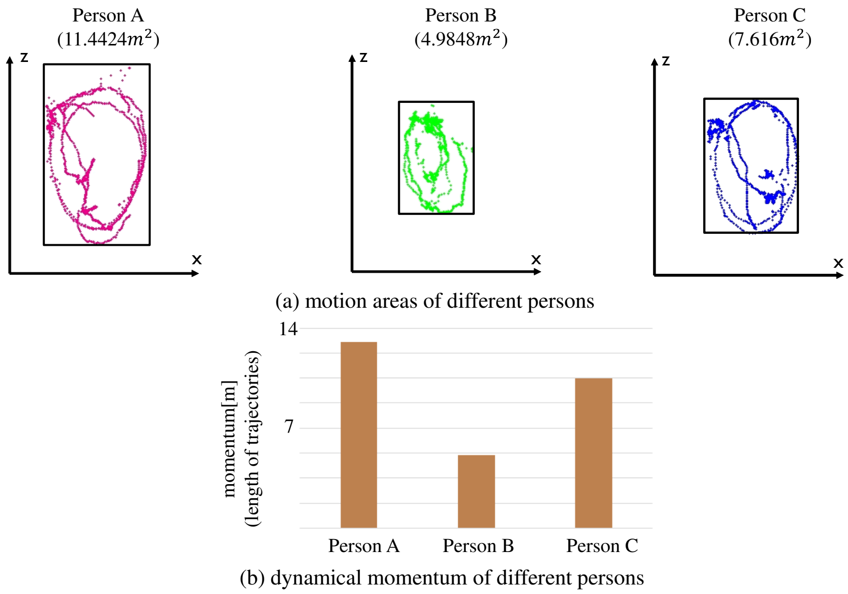


Figure 3.21: Motion areas and momentums of different persons.

(b) shows the relative distances between child 3 and the other two children. Their average relative distances during the game is 1.623m and 0.307m with the standard deviation of 0.592m and 0.235m. It can be observe that child 2 stays closer to child 3 at most of the time, and their distance is very small, even touch with each other sometimes. On the other hand, child 1 usually keeps a small distance with them. The teachers can infer their relationships that child 2 and child 3 are close friends and they like to play together. This information is especially useful for monitoring the children under natural status. By showing the children who like to play with each other, the teacher can understand the behaviors of the children better, and control the motions of the childcare robot to help to adjust the relationships among children.

3.4.6 Submission for the Application

The system of simultaneous people recognition and tracking system by using Kinect sensors is successfully applied to the childcare assisting system to track the behaviors of the children, towards the goal of assisting the nursery

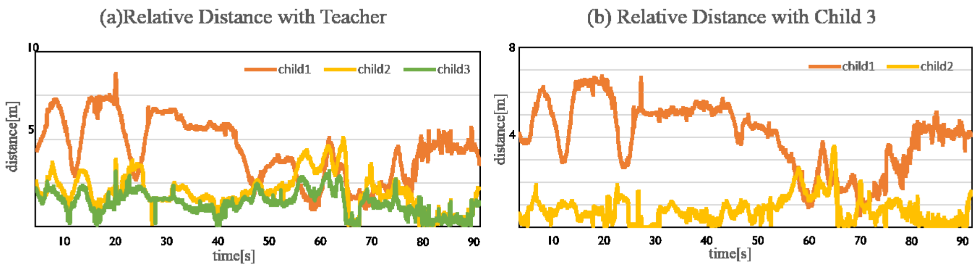


Figure 3.22: The relative distance relationships during the drum game.

teachers with the childcare work. By the proposed system, every particular child can be recognized and robustly tracked. By adding the children recognition process, and using the recognition results as the observation information to evaluate the predictions in the tracking process, the children are tracked with their ID for long period of time. The behaviors of the children are analyzed based on the requirement of the nursery teachers. The information of trajectories, motion ranges and relative distances information with the teacher and other children for each child are provided for the nursery teachers to assist the childcare work. With long term observation, the changing tendency of the children with growth can be extracted. The teachers can evaluate the progress or growth of the children with quantitative data. The system can also be used for finding out the children who does not join the class activity well, and helping to find out the autisms. However, the color information of each child cannot be repeatedly used as the children change their clothes every day. More robust personal features need to be proposed for personal recognition. Future work will also be focused on understanding different scenes by the system and provide more information to the nursery teachers.

3.5 Conclusion

In this chapter, a simultaneous people recognition and tracking by using Kinect sensors is proposed, towards the goal of providing the user status information for the guide robot. It is a general system for any people tracking task, and has been successfully applied to a childcare assisting system to track the behaviors of the children. The effectiveness of the system is evaluated by using the tour group data under lab environment. People moving in indoor environment with good illuminations have been continuously tracked more than 30 minutes, and the tracking error is never diverged. The results proved that the particular people can be continuously recognized and tracked even under crowded environment with the proposed system. The recognition algorithm keeps recognizing the detected people to improve the tracking accuracy, and the tracking algorithm records the motion history of the people and predict the positions of them to improve the recognition accuracy. Compared with conventional methods [49], the system has higher tracking accuracy and it can automatically modify the errors that caused by occlusions. That is to say, even if the system failed to track the right person, it will be modified when the particular user is recognized again. However, the color information of each person is used here, and it cannot be repeatedly used as the people may change their clothes or wear similar clothes. More robust personal features need to be proposed for personal identification. Future work will also be focused on proving the effectiveness of the system by using real guide tours information.

Chapter 4

Framework for Adaptive Motion Control

4.1 Introduction

Human-robot interaction has currently become one of the fastest growing research fields. In recent years, the development of service robots has played an important role in human-centered robotics. At present, service robots are used in offices, restaurants, hospitals, and homes [61–64]. An important function of service robots used in different fields is guiding users from one place to another [65] . With an increasing number of services provided by such robots, it is crucial to ensure service quality. Users expect the guide robot to be more sociable. However, few researchers are working on how to make the robot more sociable to improve the quality of guiding service. The guide robot cannot expect that the users will always follow it or maintain a fixed distance from it; instead, it needs to adapt to user activities and always accompany with them during the guiding process. In this chapter, an adaptive framework is proposed to control the motion of the robot to accompany the users and provide sociable guiding services. In terms of motion control, a sociable guide robot needs to carry out the following four functions: it must (1) adjust its speed to match that of the user (e.g., young people may move faster than elderly, or the same group of people may change their speed depending on their interests); (2) maintain its relationship by keeping the social distance with the users (e.g., by following them if they deviate from the guiding path); (3) prepare to restart the guiding task during “Follow” mode; and (4) take the users to their destination by the shortest path. Conventional research focuses on realizing one or more of these by changing the motion states of the robot case by case. In the work of Mizobuchi Y. et al. [66], the robot adjusts its speed to match that of the users by maintaining the relative distance, which can be updated by the voice feedback of the users' impressions. In the work of Fleury S. et al. [67], a robot that can guide users based on a stop-and-wait model was shown to be socially unappreciated. In the work of Shiomi M. et al. [68] , the robot moves backwards sometimes to maintain the guiding relationship. In the work of Oyama T. et al. [69], the museum guide robot stops and talks with the users leaving the guide tour. In the work of Pandey A.K. et al. [70], the two situations of non-leave-taking human activities and leave-taking human activities are well discussed, and the rules related to robot motion are designed to meet each condition. The authors successfully presented a way for guiding users in a socially acceptable way. However, the number of rules for this case-by-case method may diverge quickly when the environment becomes more complex. Up to now, no researcher has proposed a general framework by which all these adaptive motions of an autonomous mobile robot can be generated at the same time.

In this chapter, a framework is designed to control the robot to move adaptively in a socially acceptable way. It is the first time to consider improving the guiding quality by accompanying the users with adaptive motion control, using a single general framework rather than multiple case-by-case rules. With the proposed framework, the adaptive motions are formed automatically, and all four above-mentioned functions for a sociable guide robot can be realized. In the framework, a special artificial potential field for the users is generated, and integrated with other basic potential fields [71] generated by considering the goal that the users want to reach, in addition to the pedestrians and obstacles in the environment. The work of Nakazawa K. et al. [71] merely considered the effects from the goal and obstacles. By contrast, the proposed framework also considered the status of the users as an influence on the motion of the robot. By considering the situations of the users in this way, the proposed framework can control the robot in such a way as to adapt to the users' activities and the users can move without any restrictions. Unlike the work of Pandey A.K. et al. [70], the adaptive motions used in the proposed framework to achieve "sociable guidance" are formed naturally and automatically. The effectiveness of the framework is demonstrated by simulations, and it is observed that the users are successfully guided to their destination in a sociable way by the robot in the experiments. The aim is to design a general framework that supports adaptive motions of sociable guide robots to improve the quality of guiding services.

4.2 Design of the Framework

To guide the users in a social manner, the robot should adjust its motion to adapt to the needs of the users and support their activities. The adaptive motion control of the robot is based on the artificial potential field method [71]. The attractive and repulsive forces from the environments are generated as virtual artificial potential fields, and the motion of the robot is controlled by the gradient of the potentials. The robot moves towards the position with a lower potential. Basically, an attractive force is generated from the goal to make the robot move towards it, and a repulsive force is generated from obstacles around the robot to prevent collision. In this thesis, in order to ensure that the robot can move adaptively in a social way, the framework is designed to generate adaptive potential fields for the users and subgoal separately, and integrating them with the basic potentials from obstacles to control the robot.

4.2.1 Generation of Artificial Potential Fields

An attractive potential field is generated from the real-time updated subgoal and a special potential field generated for the user group, which may show an attractive or repulsive effect, depending on the relative distance between the robot and the users. Besides, basic repulsive potential fields are generated from the objects around the robot.

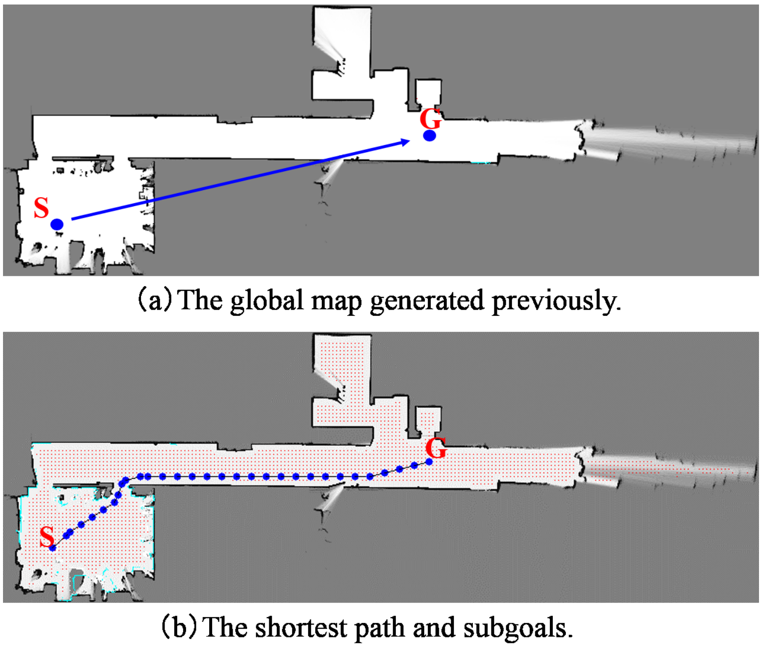


Figure 4.1: The generation of subgoals after path planning.

Attractive Potential Field From The Goal

Because the robot needs to move towards to the goal, it is needed to set an attractive potential from it. However, if only an attractive potential field from the fixed final goal is generated, dead-lock situations will occur frequently, making it difficult to integrate this field with other potential fields. Therefore, the shortest path for the robot is calculated first, and a series of subgoals are sampled from the generated path. In the environment shown in Fig. 4.1 (a), for example, the robot starts from point S and needs to move to the goal (point G). The path is generated as a series of contiguous points. Here, Dijkstra's algorithm [72] is used to calculate the shortest path. The path is updated online to manage the change in the environment. A series of subgoals is then sampled from these points, taking an interval more than the threshold d_{th}, as shown in Fig. 4.1 (b). In chapter 2, the path planning method is explained in detail. The current attractive force field is then generated from the current subgoal. The attractive force F_{ag} from the

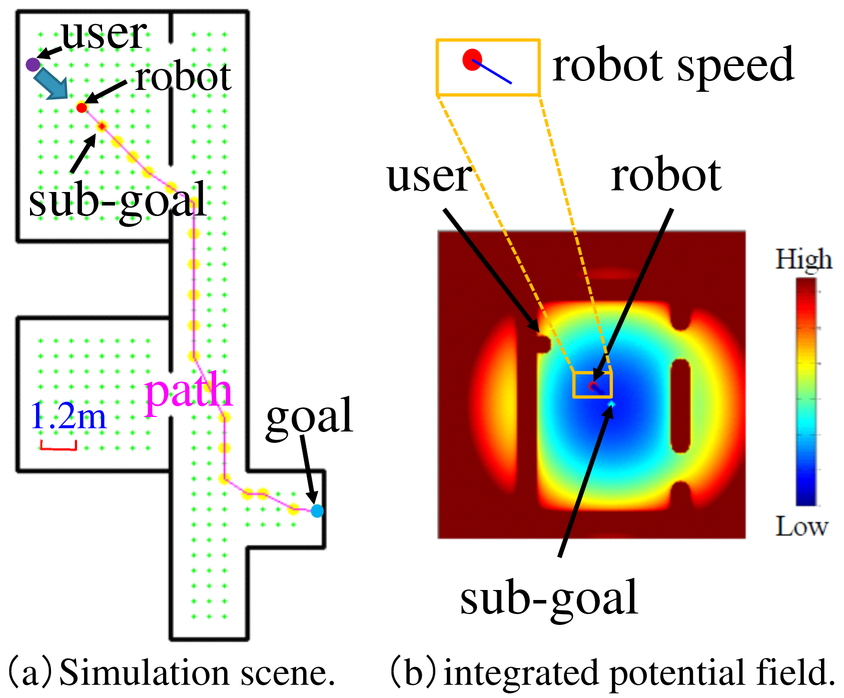


Figure 4.2: An example of guiding a user to the destination.

subgoal is set as

$$F_{ag} = k_{ag} \times d_g \tag{4.1}$$

where k_{ag} is the attraction coefficient for the goal, and d_g is the distance from the subgoal to the robot. The attractive potential P_{ag} generated is given by

$$P_{ag} = \frac{k_{ag}}{2} \times d_g^2 \tag{4.2}$$

Special Potential Field from the Users Group

The users are treated as a group and the group generates a special potential field to affect the motion of the robot. It is special as its effect may be attractive or repulsive according to the relative distance between the user group and the robot. In order to make the robot adapt to the motions of the users, an attractive effect for the users group is generated. However, this attractive potential does not start from the user group; rather, it starts from a point in front of the users group that is at a fixed distance of d_s with the users' group in the robot's direction. Here, d_s means the social distance, which is the best relative distance to maintain the guiding interaction. The attractive force F_{au} from the users' group is set as

$$F_{au} = k_{au} \times (d_s - d_u) \tag{4.3}$$

where k_{au} is the attraction coefficient for the users' group, and d_u is the distance from the users' group to the robot. The attractive potential P_{au} generated is given by

$$P_{au} = \frac{k_{au}}{2} \times (d_s - d_u)^2 \tag{4.4}$$

Repulsive Potential Field from the Obstacles

In order to prevent the robot from colliding with obstacles and pedestrians, a repulsive potential for them is set in the same way with the previous work [71] The repulsive force from each point of the objects is set as

$$F_r = \frac{k_r}{(d_r - d_0)^2}, if\ d_r > d_0\ ;\ F_r = \infty, else \tag{4.5}$$

where k_r is the repulsion coefficient for the objects, d_r is the distance from the object to the robot, and d_0 is the minimum distance to be traveled by the robot to reach the object. The repulsive potential P_r generated is given by

$$P_r = \frac{k_r}{d_r - d_0}, if\ d_r > d_0\ ;\ P_r = \infty, else \tag{4.6}$$

In the proposed framework, the repulsive force is caused from the obstacles. Normally, when the robot is far from the obstacles, these objects do not significantly influence the motion of the robot. When the robot is close to the obstacles, however, the latter's influence should be sufficiently high to prevent the robot from colliding with the obstacles. The repulsive force is set in quadratic inverse mode as shown in Equation (4.5) for ensuring the obstacles will not affect the robot motion much when they are far way and the robot can stop before colliding with the obstacles under a high speed.

4.2.2 Adaptive Guide Robot Controlled by Integrated Potential Fields

The coefficients for different potentials are set as constant values to ensure that the framework is generalizable and available for different users under different situations. All the generated potentials are integrated together to control the robot. In the case of the scene shown in Fig. 4.2 (a), the robot tries to guide the user to the goal, and the user is following the robot. The integrated potential field is shown in Fig. 4.2 (b). The robot will move towards the position with the lowest potential, and hence, towards the subgoal. The proposed adaptive potential fields from the users' group and the subgoal make the robot guide the users in a manner that is more sociable. Note that the attractive force from the users' group may have an attractive effect or a repulsive effect from the perspective of the robot. When the distance between the robot and the users' group d_u is smaller than d_s (here, the social distance is the most comfortable relative distance that can be set in advance), the users are getting close to the robot, and the influence of the users' group will be the same as that of the subgoal to force the robot to move more quickly. Moreover, the closer the users move toward the robot, the greater the influence. Naturally, the robot will adjust its speed to maintain the best social distance. On the other hand, when the relative distance d_u is greater than social distance d_s, the users are moving away from the robot, and the influence of the users' group will be opposite to that of the subgoal. The robot will thus move slower. When the potential from the users' group is stronger than that generated from the subgoal, with the users group moving further away, the robot will naturally move towards the user group. From the result of the motion, the robot has naturally changed to "Follow" mode. Only when the users get closer to the robot again and the attractive effect from the subgoal becomes stronger, the robot will move towards the subgoal. The robot judges that the users' group has returned in this way, and restarts the "guide" mode. During the entire guiding process, the shortest path is updated online such that the subgoal is likewise updated online, even if the robot is in "Follow" mode. With the influence from the real-time updated subgoal, the robot stays in the best position for returning back to "Guide" mode when it follows the users. In this way, the robot guides the users along the shortest path. Notice that these speed adjustments and motion mode changes are carried out naturally by the generated adaptive potentials, rather than simply defined by some if-then strategies for the robot. In this way, the robot can (1) adjust its speed to meet the users' intent, and (2) alternate between "Follow" and "Guide" modes automatically to maintain the guiding relationship. Even if the robot is in the "Follow" mode, the shortest path is updated online, and the subgoal is also updated. The robot will not only follow the users, but also stay in the best position for returning to "Guide" mode. Whenever the users restart the original task, the robot will navigate in the direction of the goal. In this way, the robot can (3) prepare for returning to the guiding task from the best position, such that the robot (4) takes the users to the destination via the shortest path. Therefore, these four functions are all realized with the proposed framework, without needing to set several motion rules case-by-case. However, in complex environments, deadlock problems may occur with this integrated potential method when multiple positions with a local minimum potential are generated. This problem is solved by the temporary use of the Laplace potential to control the robot [71].

4.3 Simulation

The purpose of the proposed framework is to guide the users from their current position to a desired place, whether the users are cooperative or not, in a socially acceptable way. Simulation is processed first to check the effectiveness of the proposed framework. In the simulation work, the motion of a user to be guided is controlled by using a game pad, and whether the robot can provide social guiding services under different kinds of situations is checked. The user was free to move in any direction at any time, stop or move at any moment, and even deviate from the guided path and move to any other place. The speed of the user could also vary. The same situation is repeated for more than 50 times to check the reproducibility of the proposed framework. The simulation environment was built with Visual Studio 2013

Table 4.1: The values of coefficients in simulation

$k_r = 0.2$	$k_{ag} = k_{au} = 2$	$k_{air} = 0.24$	$d_0 = 0.3$m
$d_s = 1.2$m	$g = 10m/s^2$	$\mu = 0.01$	$m = 2$kg

on a note-PC (HP OMEN15-5100, CPU 2.6GHz, RAM 8GB, 64bit Windows System). OpenCV libraries were used to generate the figures and the appearance of the potential field. We simulated an indoor scene with two rooms and a corridor, as shown in Fig. 4.2 (a), in which one point represented 35mm in the real world. The robot can be seen as a point in the image. In the simulation work, the influence of friction and air resistance are also considered. The resultant force F_{result} of the robot is given by

$$F_{result} = F_{ag} + F_r + F_{au} - \frac{v}{|v + 0.01|} \times (k_{air}v^2 - \mu mg) \tag{4.7}$$

Here, k_{air} is the air resistance coefficient, v is the speed of the robot, μ is the frictional coefficient, and g is the acceleration of gravity. The values of the variables are shown in Table 4.1. The values of k_{air}, μ, m are set as the real properties of the robot. In particular, d_s, d_0 is decided by the human social distance [73] and minimum distance to stop before colliding with its maximum deceleration. The attractive force from the subgoal should be more or at least equal to the sum of the air resistance and friction when the influence from the users is ignored. Thus, k_{ag} can be calculated, and k_{au} is set equivalent to k_{ag} to ensure that the force from the users has the same changing rate as the attractive force from the subgoal. Next, k_r is decided by ensuring that the robot is not significantly influenced by obstacles that are far away. Thus, the robot can reach maximum deceleration as it approaches the obstacles.

4.3.1 Guiding a user who changes his speed many times

The task is set as guiding a user from the start point to the goal by leading the user out of a room and passing through the corridor, as shown in Fig. 4.2 (a). The user adopted different kinds of uncooperative motions during the process, and the reactions of the robot are checked to observe whether the robot can provide sociable guiding services. Fig. 4.3 shows the motion trajectories of the user and the robot. Eight snapshot points are taken to study the reactions of the robot. The integrated potentials and the resultant forces on the robot are shown in Fig. 4.4. The color changes from blue to red to represent the potential changes from low to high, and the red points with blue lines show the positions and speeds of the robot. The robot moves in the direction of the blue lines, and the length of the blue lines represents the magnitude of the resultant forces. The changing tendencies of the robot's speed (pixel/frame) and the relative distance (pixel) between the robot and the user are shown in Fig. 4.5. As shown in Fig. 4.5 (a), the speed of the robot changed between plus and minus values. Here, a plus speed means that the robot was moving toward the goal, and a minus speed means that the robot was moving toward the opposite direction of the goal.

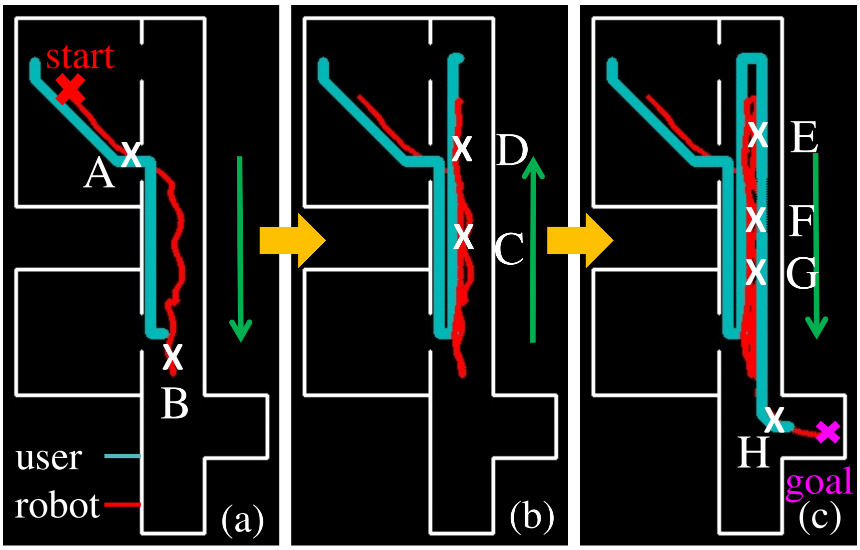


Figure 4.3: Motion trajectories of the user (light blue) and the robot (red) and snapshot points (point A-H).

(a) Speed Adjusting Test 1 The user followed the robot first for a while (Fig. 4.3 (a)), suddenly accelerating (A in Fig. 4.3 (a)), and then

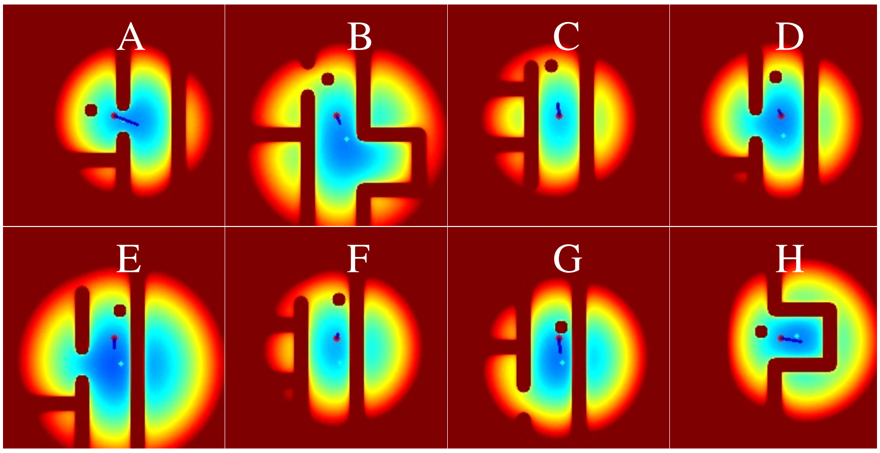
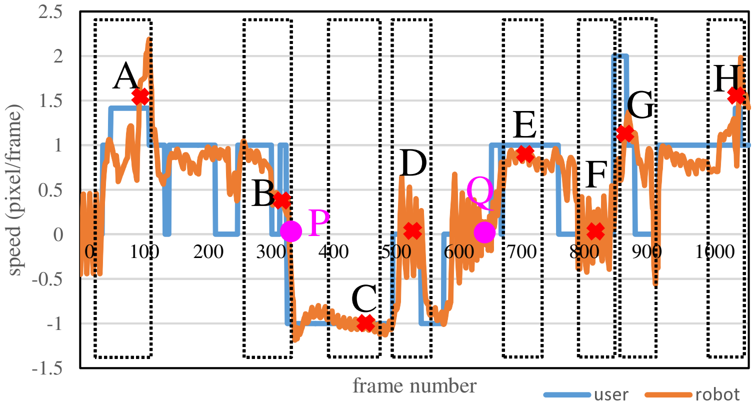


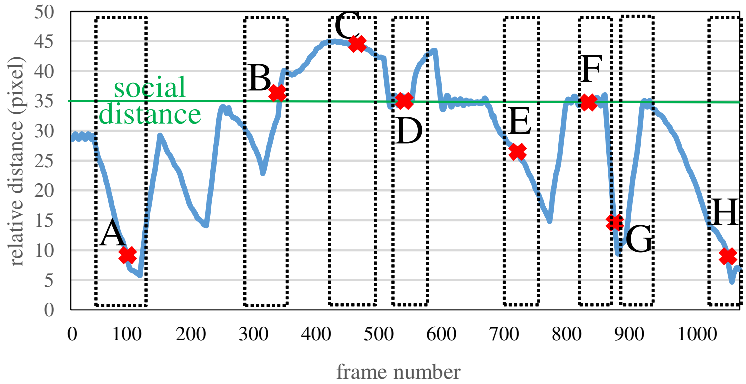
Figure 4.4: Integrated potentials and resultant forces (blue lines) on the robot (red point).

moving forward suddenly (B in Fig. 4.3 (a)). As it is expected, the robot also accelerated (A in Fig. 4.5 (a)) when the user came close to the robot (A in Fig. 4.5 (b)), as the resultant force on the robot increased (A in Fig. 4.4). The robot slowed down (B in Fig. 4.5 (a)) when the user suddenly stopped going forward and the relative distance was greater than the social distance (B in Fig. 4.5 (b)) as the resultant forces on the robot decreased. It was observed that the robot correctly adjusted its speed to match that of the user.

(b) Relationship Maintaining Test The robot then tried to leave the original path and move backward (Fig. 4.3 (b)), during which the user maintained his speed (C in Fig. 4.3 (b)) and stopped once (D in Fig. 4.3 (b)). The robot changed its direction of motion (P in Fig. 4.5 (a)) to follow the user. It first accelerated in the direction of the user (C in Fig. 4.5 (a)). When the user went backward, the relative distance increased (C in Fig. 4.5 (b)), as the resultant force on the robot changed in the user's direction and increased further (C in Fig. 4.4). The robot also stopped (D in Fig. 4.5(a)) when the user stopped and their relative distance decreased to social distance (D in Fig. 4.5 (b)), as the resultant force on the robot was almost zero (D in Fig. 4.4). After a while, the user returned to the guiding process and



(a) Moving speeds of user and robot with increasing time



(b) Relative distance between user and robot with increasing time

Figure 4.5: Changing tendency of vectors and relative distance.

followed the robot again (Fig. 4.3 (c)), and the robot changed back to the guiding mode at point Q in Fig. 4.5 (a). It was observed that the robot automatically formed "Guide" and "Follow" modes and switched between them to accompany the user. By analyzing the entire changing tendency of the robot's speed in Fig.4.5 (a) and the relative distance in Fig. 4.5 (b), it was found that the robot tried to follow the user's moving pace and maintain the guiding relationship by maintaining the social distance, even when the user tried to leave the original path.

(c) Speed Adjusting Test 2 The user then moved at a stable speed in order to follow the robot (E in Fig. 4.3(c)) before stopping suddenly (F in Fig. 4.3 (c)). The robot also accelerated to adapt to the user's speed (E in Fig. 4.5 (a)) when the user was getting close (E in Fig. 4.5(b)) and the resultant force on the robot increased in the subgoal direction (E in Fig. 4.4). The robot also stopped to wait for the user (F in Fig. 4.5(a)) when the relative distance increased such that it was equivalent to the social distance (F in Fig. 4.5 (b)). The resultant force on the robot reduced to almost zero (F in Fig. 4.4). Subsequently, the user changed speed twice to observe the reaction of the robot (G, H in Fig. 4.3 (c)). The robot reacted correctly as it accelerated (G, H in Fig. 4.5 (a)) when the relative distance decreased (G, H in Fig. 4.5 (b)). These accelerations were caused by the changes in the integrated potential and the resultant force on the robot (G, H in Fig. 4.4). Again it was observed that the robot correctly adjusted its speed to match that of the user.

4.3.2 Guiding a user who drops by at many places

Another scene was simulated where the user dropped by at multiple places while the robot followed him. Figure 4.6 shows the adaptive motions of the robot.

(d) Preparation for Restarting Guiding Task Test The user first followed the robot for a while (Fig. 4.6 (1)-(a)), and suddenly went backward and moved around in the room (Fig. 4.6 (1)-(b)). Then, the user went back to the guiding task and followed the robot to the goal (Fig. 4.6 (1)-(c) and -(d)). Figure 4.6 (2) shows how the robot updated the path for each frame, even in "Follow" mode. The integrated potential fields are influenced by the position of the user, objects in the environment, and the current subgoal. As the path is updated online, the subgoal is also updated, ensuring that the robot stays in the best position for returning back to "Guide" mode when it follows the user. The integrated potential is shown in Fig. 4.6 (3), and the motion of the robot is controlled by the resultant force. From the motion trajectory in Fig. 4.6 (1), it was found that the robot was not only following the user when the latter dropped by at several places, but also prepared for returning back to the guiding task, as its trajectory is almost a straight line, differing from the user's trajectory (shown in Fig. 4.6 (1)-(b)). Thus, whenever the user wants to return, the robot stands by in the goal direction. The generated paths are the shortest in the environment, and the robot tries to return in the shortest path. These two strategies ensure that the user is guided to the destination in the shortest path.

The results were compared with a conventional method [71] in which the robot tried to keep a constant distance with the user by merely considering the effect from the user and obstacles in "Follow" mode. Figure 4.7 shows the motion of the robot using this conventional method. The task was the same and the user adopted the same motions. After following the robot for a while (Fig. 4.7-(a)), the user suddenly moved backward and then moved around in the room (Fig. 4.7-(b)). After that, the user went back to the guiding task and followed the robot to the goal (Figs. 4.7 (c) and 4.7 (d)). The pink line in Fig. 4.7 shows the motion of the robot. It can be observed in Figs. 4.7 (b) and 4.7 (c) that the robot followed the user to move around in the room in "Follow" mode, and the user needed to wait for the robot to adjust its motion to the guiding path when the user moves back to the robot. The average path length of the proposed method was 62 pixels (i.e., 2.17 m), less than that of the conventional method, with a standard deviation of six pixels (i.e., 0.21 m). Similar tasks were next carried out, in which an uncooperative user left the original path multiple times in different ways. All the results show that the path lengths generated by the proposed method were shorter, and these differences became more apparent when the user moved more uncooperatively. Thus it is demonstrated that the proposed method ensures that the robot guides the user to the destination via the shortest path.

4.3.3 Guiding a cooperative users' group

The simulation works of guiding a user to move around have already proved that the proposed framework can adapt to different kinds of motions. The situations for guiding a cooperative users' group are similar with guiding one user. In this case, the users form a users' group and the average position of

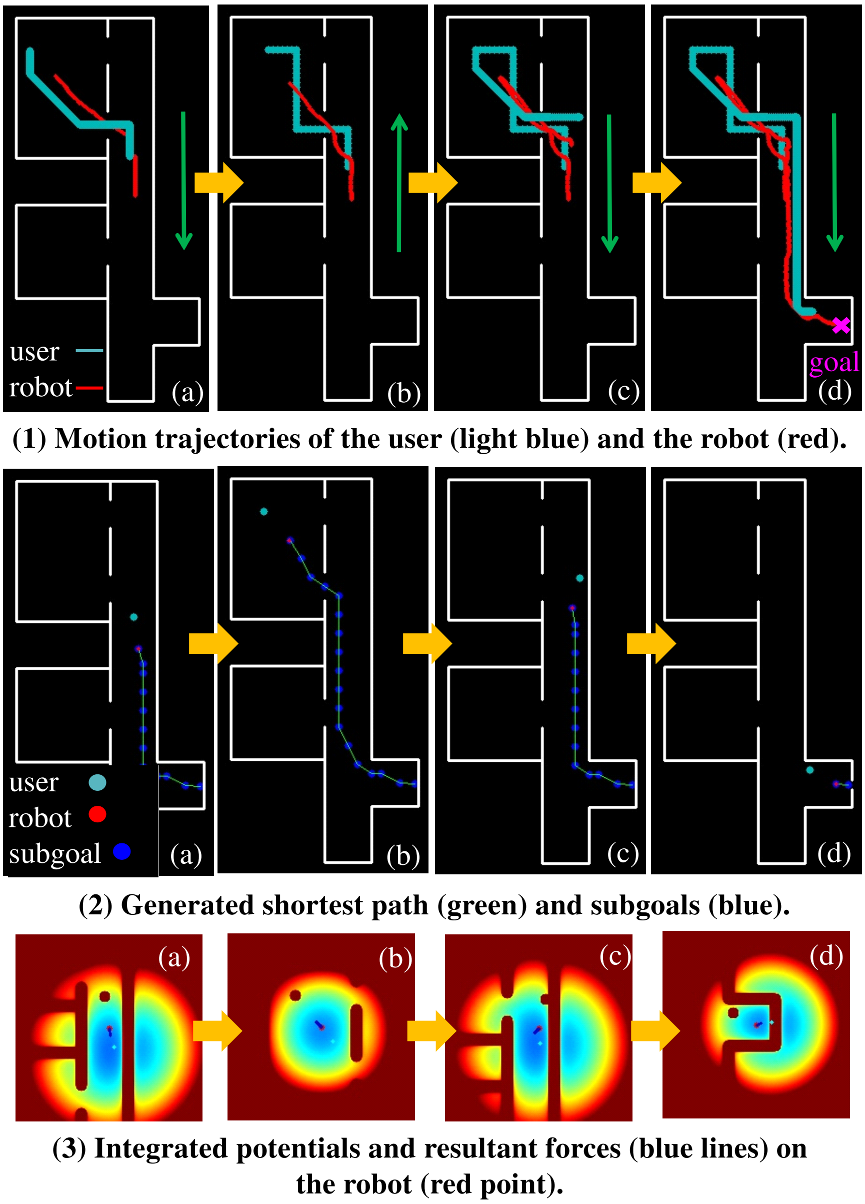
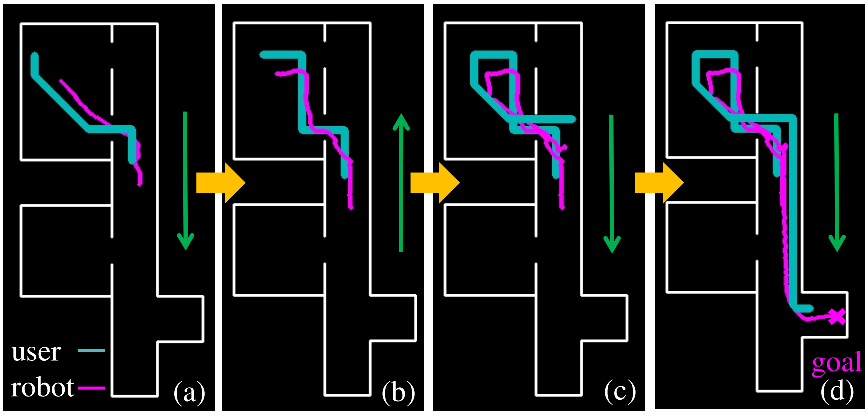


Figure 4.6: Adaptive motions for guiding an uncooperative user.

all the users can be used as the position of the group. The group position generates the special potential field and affects the motion of the robot. Fig. 4.8 shows the motion trajectories of the users' group, which is formed from



Motion trajectories of the user (light blue) and the robot (pink).

Figure 4.7: Guiding path by using the conventional method.

three users, and the robot. During the task, the users were controlled to follow the robot with similar speeds, which contains a small noise to make their motion more like real human beings. The integrated potentials and the resultant forces on the robot are shown in Fig. 4.9. The users followed the robot together for a while (Fig. 4.8 (a)) in the beginning, and the robot moved forward with a high speed as the users' group formed a repulsive force to push the robot move faster (Fig. 4.9 (a)). In Fig. 4.8 (a), the light blue line shows the trajectory of the user group, and the trajectory of the robot is shown by the pink line. The real trajectories of the users are shown as the colored lines around the group trajectory. Actually the group position just showed the center of the users, and its trajectory is a virtual line. However, the robot treated the users as a group so that it equaled with the situations that all the users move together in their center position. Then the users slowed down and stopped for a while (Fig. 4.8 (b)), and the robot also slowed down and waited for the users as expected (Fig. 4.9 (b)). Then the users started to move forward the robot again (Fig. 4.8 (c)) , the robot accelerated and guided the users' group to continue the task (Fig. 4.9 (c)). The robot moved smoothly to the destination while adjusting its speed to adapt the users' group, showing as waited the users for a few times.

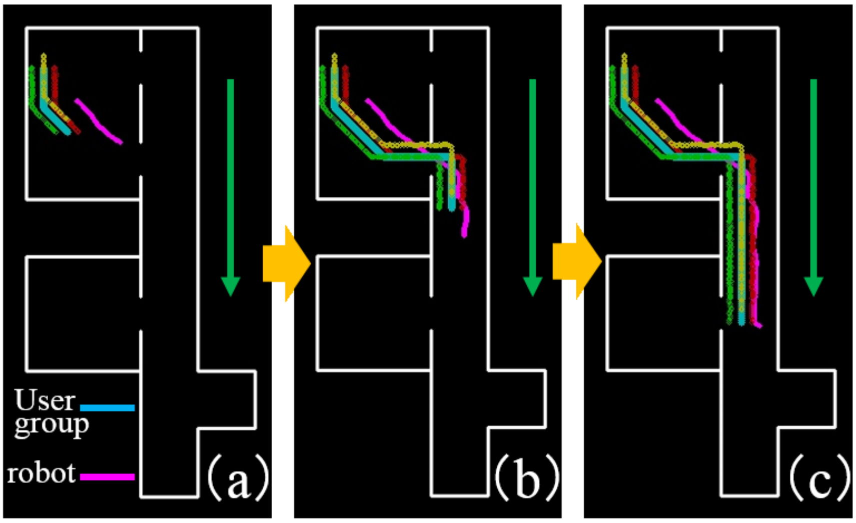


Figure 4.8: Motion trajectories of the user (light blue) and the robot (red) for guiding a cooperative users' group.

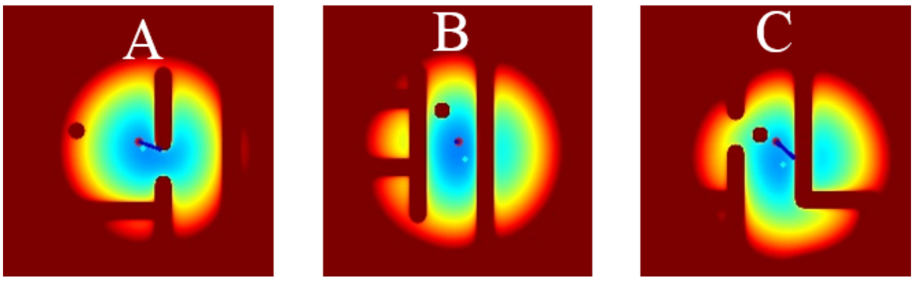


Figure 4.9: Integrated potentials and resultant forces (blue lines) on the robot (red point) for guiding a cooperative users' group.

4.3.4 Guiding a users' group in which someone is not cooperative

For most of the cases, the users may be cooperative to the robot since the social ability of human beings makes them tend to move together. However, as the role of proving services, the social guide cannot restrain the motion of the users, and should be able to deal with the problems that a few users are not cooperative. Fig. 4.10 shows the motion trajectories of the user (light blue) and the robot (red) for guiding a users' group in which someone is not cooperative. The integrated potentials and the resultant forces on the robot are shown in Fig. 4.11. During the guiding process, the users similarly formed a group, and the average position of the group generated the special potential field to control the motion of the robot. However, the uncooperative user may leave the group and he/she may joined the group again after a while. In Fig. 4.10 (a), the light blue line shows the trajectory of the users' group, and the trajectory of the robot is shown as the pink line. The real trajectories of the users are shown as the colored lines around the group trajectory. The dark red line shows the average position of the main members, and the yellow line shows the motion of the uncooperative user. In the beginning, the uncooperative user also moved towards the robot, and he/she could be considered as a group member. Thus, the average position of the whole group was calculated from all the users, shown as the light blue line is in the middle of the dark red and yellow lines. As the main members had higher weights, the average position was closer to the main members. The robot moved forward with a high speed since the users' group formed a repulsive force to push the robot move faster (Fig. 4.11 (a)). Then the uncooperative user stopped as shown in Fig. 4.10 (b). As the uncooperative user was not so far away from the main members, he/she was still judged as a group member and the robot slowed down to wait for the user (Fig. 4.11 (b)). After that, the uncooperative user started to move backward, keeping getting away from the main group members. When the distance was bigger than the group forming threshold $D_{group_{th}}$ that was set advance, the user was judged as leaving the group so that the average position of the group was equal to the average position of the main members, showing as the dark red line was overlapped with the light blue line (Fig. 4.10 (c)). The robot changed to service for the main members, who formed a small group, and started to move again (Fig. 4.11 (c)). After a while, the disappeared uncooperative user came back to the robot again, and the robot recognized him/her as a previous user so that the robot counted him/her as a group member again (Fig. 4.10 (d)). As the user was behind the main members, the robot even changed its motion direction and moved backward for a second to wait for the user (Fig. 4.11 (d)). With time went on, the uncooperative user came back to the group and moved together with the group, showing as the average position of the group (light blue line in Fig. 4.10 (e)) was in the middle of the main members and the uncooperative user (dark red and yellow lines in Fig. 4.10 (e)). The robot accelerated again to adaptive the speed of the users (Fig. 4.11 (e)). When the uncooperative user left the group again (Fig. 4.10 (f)), the robot reacts in the same way to change to service for the small group (Fig. 4.11(f)), showing as the average position of the group was overlapped with the main members' position (light blue line was overlapped again with the dark red line in Fig. 4.10 (f)).

From the results, it is observed that the robot adequately accompanied the users when guiding them to the destination. The adaptive motions are automatically generated by the proposed framework. From the tests, it is observed that the robot will (1) adjust its speed to match that of the user; (2) maintain their relationship by maintaining the social distance with the users; (3) prepare to restart the guiding task in "Follow" mode; and (4) take the users to their destination via the shortest path. From the perspective of motion control, our robot works well as a sociable guide robot. Thus, it was proved that the proposed framework worked satisfactorily for adaptively controlling the motion of the robot and providing sociable services. This simulation results also help to decide the coefficients when controlling a real robot.

4.4 Experiments

The efficiency of the proposed framework was confirmed through different guiding tasks, using the robot shown in Fig. 4.12. The mobile platform PIONEER3-DX is used, which is manufactured by MobileRobots. A forward-looking laser range finder (LRF) at a height of 32 cm, and a backward Kinect sensor at a height of 100 cm were attached to the robot. The screen of the computer showed the distances to the goals on the screen to remind the users about the tasks. The motion of the robot was controlled by integrated poten-

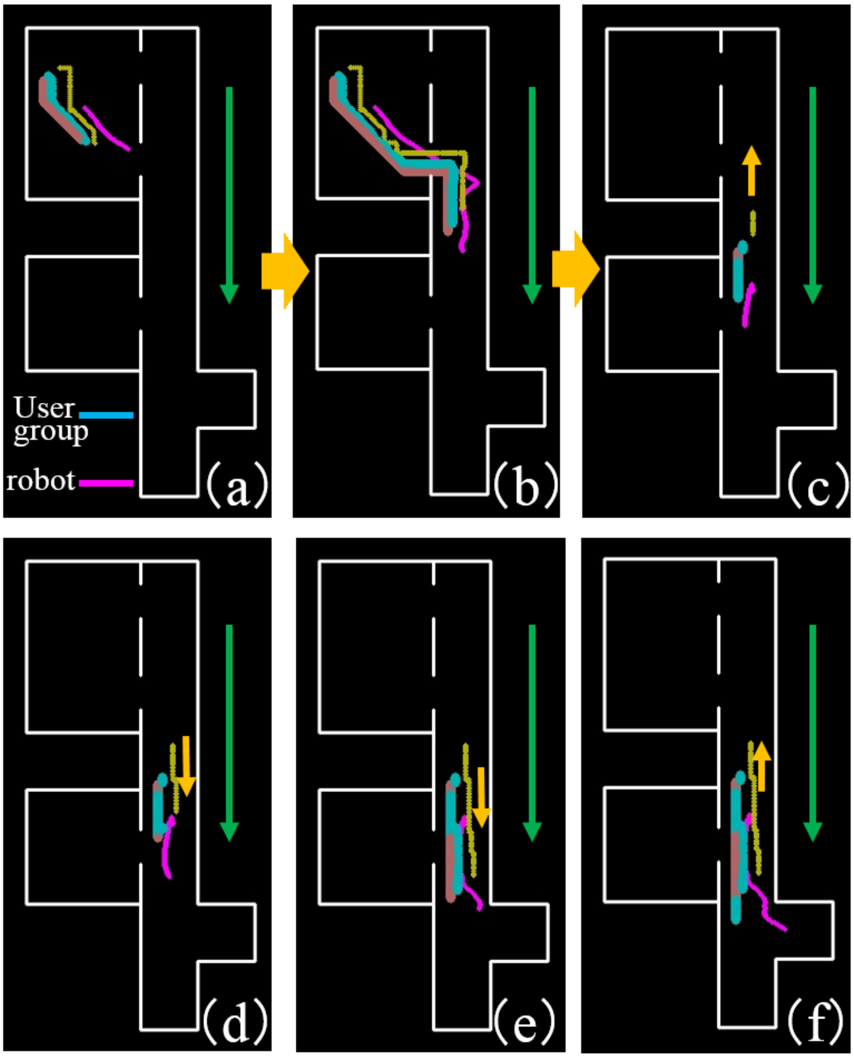


Figure 4.10: Motion trajectories of the user (light blue) and the robot (red) for guiding a users' group in which someone is not cooperative.

tials, and a threshold F_{th} was set for the resultant force, in order to prevent overreactions. Here, F_{th} has the same function as it does with friction, in order to keep the robot more stable. The resultant force F_{result} is given by

$$F_{result} = F_{ag} + F_r + F_{au}, if \ |F_{result}| > F_{th}; F_{result} = 0, else \tag{4.8}$$

Because the speed of the wheels for the real robot can be directly con-

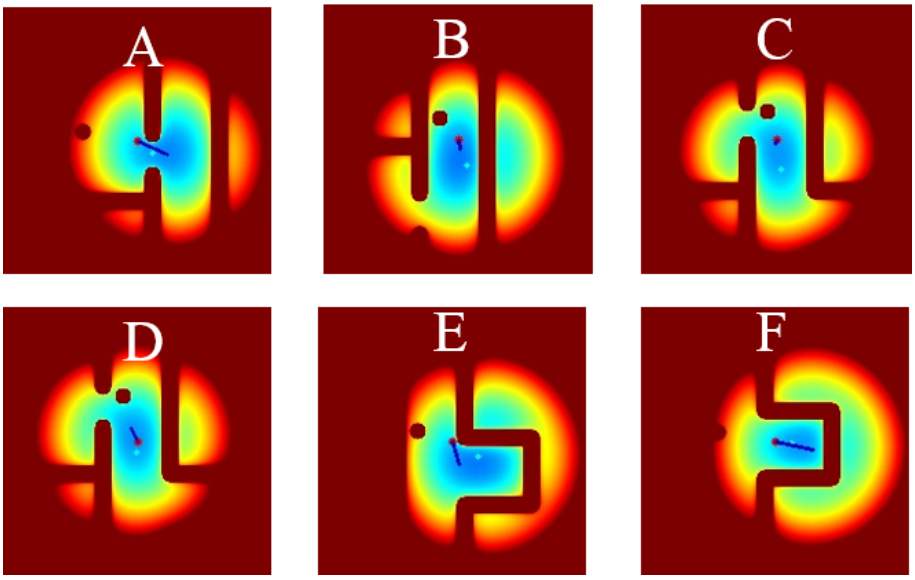


Figure 4.11: Integrated potentials and resultant forces (blue lines) on the robot (red point) for guiding a users' group in which someone is not cooperative.

Table 4.2: The values of coefficients for real robot.

$k_r = 0.2$	$k_{ag} = k_{au} = 2$	$d_0 = 0.3\text{m}$
$d_s = 1.2\text{m}$	$m = 2\text{kg}$	$F_{th} = 2.34$

trolled, the speed of the robot was calculated by:

$$v = v_{cur} + F_{result}t/m \tag{4.9}$$

where v_{cur} is the current speed of the robot. The values of the variables are shown in Table 4.2. These variables were set by referring the values in the simulation work.

The experiments were conducted in an indoor environment, as shown in Fig. 4.1 (a). Any position in the map can be chosen as the starting point or the goal, and the task of the robot was set as guiding the users from point S to point G as an example.

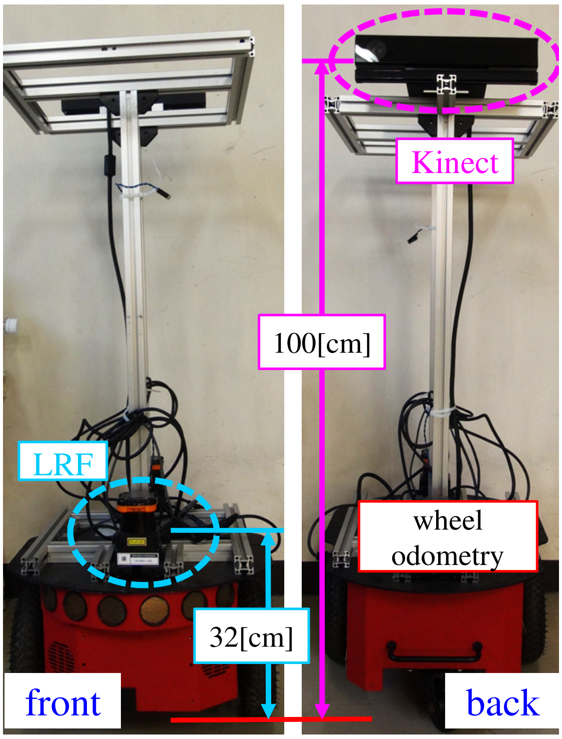


Figure 4.12: The view of the guide robot.

4.4.1 Guiding a user to the destination

One user each time was guided to the destination without any restrains; the user was only informed that he would be guided by the robot to a place, and this task was repeated with 20 different adult users.

Guiding a cooperative user to the destination

Most of the users (16 participants from a total of 20) were cooperative, and the robot took them to the destination smoothly. In the beginning of the task the users were close to the robot, and they decided when to initiate the experiment by beginning to move. The users started to move at their will and the average relative distance they felt comfortable to move can be calculated. The result is around 1.2m, which also proved that the social distance was set properly. The 16 cooperative participants moved at different speeds, and some users changed their speed during the guiding task.

(a) Speed Adjusting Test The guide robot adaptively adjusted its speed to meet the needs of each user. The robot almost maintained the social distance, regardless of how fast the user moved. It is observed that the robot correctly adjusted its speed to match that of all the 16 cooperative users.

Guiding an uncooperative user to the destination

2 representative users were selected from the other 4 uncooperative users who made the situations more complex, and studied the adaptive motion of the robot in detail to show the usefulness of the proposed framework.

(b) Relationship Maintaining Test The guiding process is shown in Fig. 4.13, and the robot trajectory is shown in Fig. 4.14. The user was normally guided by the robot (Fig. 4.13 (a)), but he suddenly stopped (Fig. 4.13 (b)). Then, he moved in opposite direction, as he was interested in the posters on the wall (Fig. 4.13 (c)). After reading the posters (Fig. 4.13 (d)), the user returned to the guiding task (Fig. 4.13 (e)) and followed the robot to the destination (Fig. 4.13 (f)). The robot first guided the user from point A to B (Fig. 4.14 (a)) normally. Then, the robot slowed down and stopped when the user stopped. After that, the robot changed its direction to follow the user (point B to C in Fig. 4.14 (b)) as the user started moving backward. After the user returned to the robot, the robot changed its direction again, and guided the user to the destination smoothly (point C to D in Fig. 4.14 (c)). It was observed that the robot automatically switched between "Guide" and "Follow" modes to accompany the user. The robot tried to follow the user's moving pace and maintain the guiding relationship by keeping the social distance, even when the user tried to diverge from the original path.

(c) Preparation for Restarting Guiding Task Test Another guiding process is shown in Fig. 4.15, and the robot trajectory is shown in Fig. 4.16. Normally, the user was first guided by the robot (Fig. 4.15 (a)). Then, the user moved back and walked around in the room (Fig. 4.15 (b)-(d)). After some time, the user returned to the guiding task (Fig. 4.15 (e), (f)) and followed the robot to the destination. The robot first guided the user from

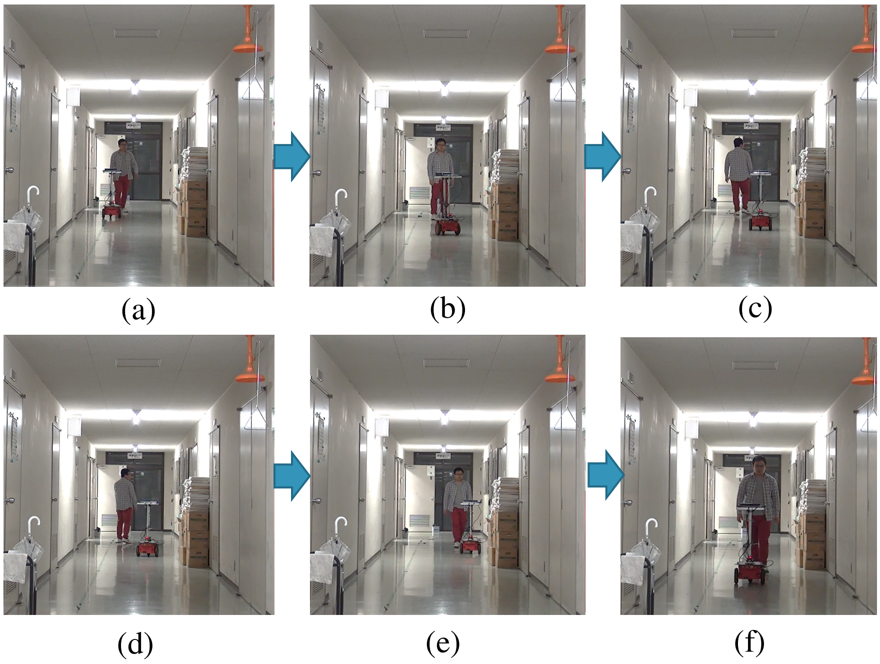
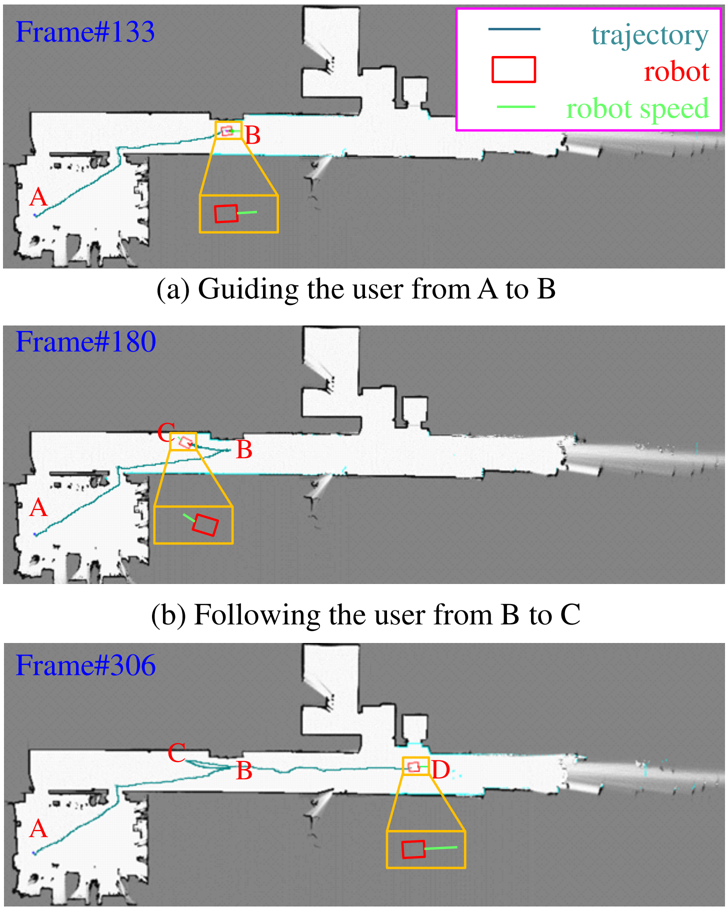


Figure 4.13: Scenes when the robot guided an uncooperative user who walked backward.

point A to B (Fig. 4.16 (a)) normally. Then the robot changed its motion to follow the user (point B to C in Fig. 4.16 (b)) as the user started moving backward and walked around in the room. Notice that in “Follow” mode, the robot almost moved in a straight line, despite the user walking around and stopping at multiple places in the room. The robot did not merely follow the user, but also managed to stay in the subgoal direction while in “Follow” mode, ready to restart the guiding task (Fig. 4.15 (b)-(d)) from the place that requires the least effort to arrive at the original goal. These adaptive motions ensure that the robot guided the user along the shortest path. After the user returned to the robot, the robot again changed its direction of motion, and guided the user to the destination smoothly (point C to D in Fig. 4.16 (c)). It was observed that the robot prepared to restart the guiding task while in “Follow” mode by always standing in the direction of the goal, and the robot took the users to their destination via the shortest path by updating its path



(a) Guiding the user from A to B

(b) Following the user from B to C

(c) Guiding the user from C to D

Figure 4.14: Motion trajectories of the robot when guiding an uncooperative user who walked backward.

online.

The results of the experiments matched well with the simulation results, proving that the proposed framework works for providing socially acceptable services.

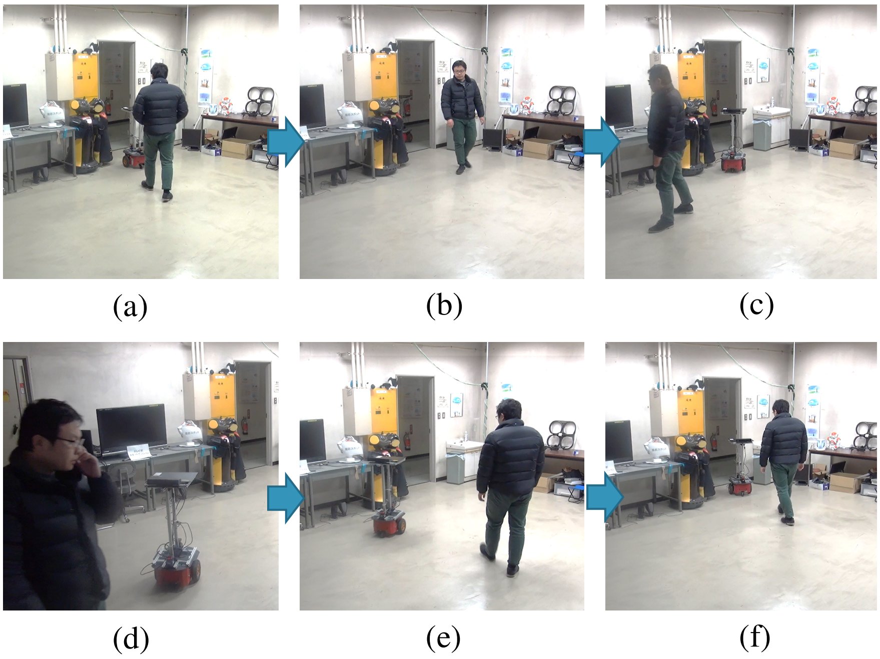


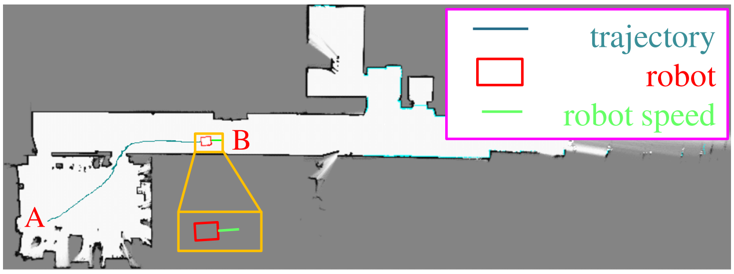
Figure 4.15: Scenes when the robot guided an uncooperative user who walked around the room.

4.4.2 Guiding a users' group to the destination

Multiple users each time were guided to the destination as a group; the users' group was only informed that they would be guided by the robot to a place.

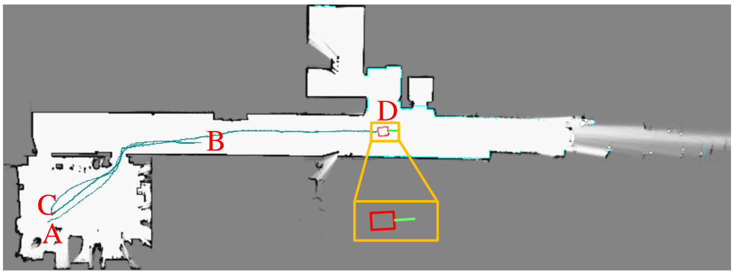
Guiding a cooperative users' group to the destination

To show that the proposed framework is also effective when guiding a group of people, the robot was made to guide different groups of users to the destination without placing any restrictions on them. These user groups tended to be more cooperative, because they usually take care of each other. The robot treats the users as a group, whose center is the mean position of all the users. The integrated potential field is similar to the potential field generated for a single user. 20 people were asked to form different groups (10 different groups with 3 to 5 people) for the experiments. An example of this



(a) Guiding the user from A to B

(b) Following the user from B to C



(c) Guiding the user from C to D

Figure 4.16: Motion trajectories of the robot when guiding an uncooperative user who walked around the room.

scene is shown in Fig. 4.17. The figure shows that the user group moved at a relatively stable speed, and that the robot moved at a similar speed in front to guide them. The robot trajectory is shown in Fig. 4.18, demonstrating that the robot successfully moved to the goal point.

The robot smoothly guided all the 10 user groups to the destination. **(a) Speed Adjusting Test** The guide robot adaptively adjusted its speed to meet the needs of the users'. The robot almost maintained the social

Figure 4.17: Scenes when the robot guided a cooperative users' group.

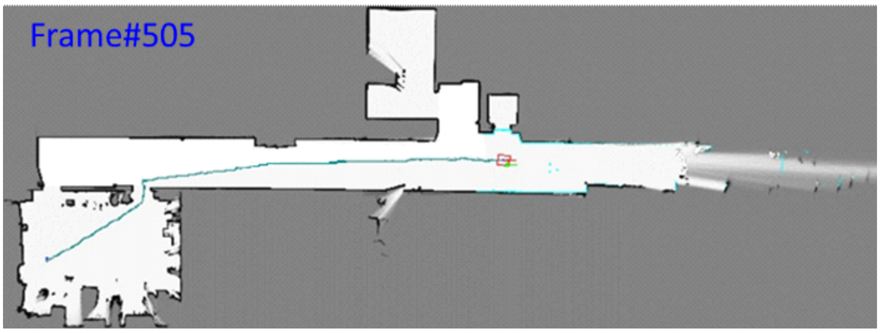


Figure 4.18: Motion trajectories of the robot when guiding a cooperative users' group.

distance, regardless of how fast the users moved. It is observed that the robot correctly adjusted its speed to match that of the users groups.

Guiding a users' group to the destination in which someone is not cooperative

To prove that the proposed framework can also adapts to the users' group even if someone is not cooperative, the robot was made to guide different groups of users, some of whom were asked to be uncooperative. The others still moved without any restrictions. During the tasks, one uncooperative user was asked to leave the group once and return to the group after a while. In the beginning ,the robot tracked all the members in the group and guided them to move forward. When the uncooperative users started to move away, the robot warned the uncooperative user once before being judged as leaving the group. The robot reacted similar with the simulation work. When a user tends to be uncooperative and tries to leave the group, the average position of the users' group will be affected by the uncooperative user who tried to leave, and the robot adjusted its speed to wait for the user. As most of the users keep moving forward and the uncooperative user left far away from the main members, the robot reacted as service the main members, and deleted the uncooperative user from the user group. The robot guided the small group to the destination. After a while, the disappeared uncooperative user came back to the robot again, and the robot recognized him as a previous user so that the robot counted him as a group member again. As the user was behind the main members, the robot waited for the user until he got close to the robot. Then the robot guided the users' group to the final destination.

(b) Relationship Maintaining Test The guide robot tried to accompany with the users' group during the guide task. When one of the users tried to leave the group, the robot warned him once to try to main the relationship with him besides the other. After his leaving, the robot accompanied with the left members while waiting for their coming back. When a new person was detected, the system would judge whether the new person was the original user. Once confirmed, the robot tired to adapt to all the members.

Guiding a users' group to the destination which drops by at other places

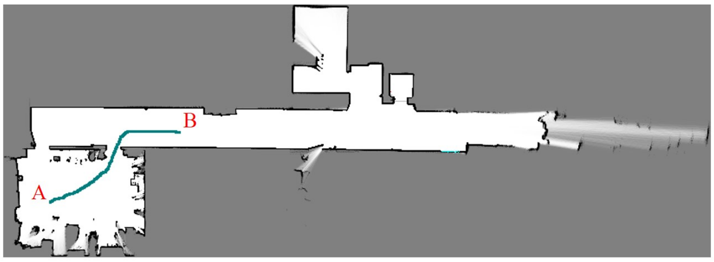
The proposed framework can also adapts to the uncooperative users' group. During the guide task, the users' group deviated the original path once, and dropped by the place that they were interested in behind them. In the beginning ,the robot tracked the users' group and guided them to move forward. When the users started to deviate the original path together, the robot decelerated and then changed its motion direction and followed the users for a while. After a while, the users came back to the robot, and the robot changed back to the "guide" mode.

(b) Relationship Maintaining Test The guide robot tried to accompany with the users' group during the guide task. When the users tried to leave the original path, the robot changed its motion direction and followed it. After they moving towards the guide robot again, the robot changed back to "guide" mode.

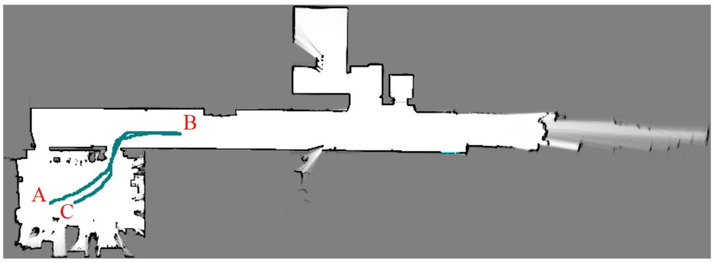
(c) Preparation for Restarting Guiding Task Test The robot trajectory is shown in Fig. 4.19. The robot first guided the users' group from point A to B (Fig. 4.19 (a)) normally. Then the robot changed its motion to follow the users' group (point B to C in Fig. 4.19 (b)) as the user started moving backward and walked around in the room. Notice that in "Follow" mode, the robot almost moved in a straight line, despite the users' group walking around and stopping at multiple places in the room. The robot did not merely follow the user, but also managed to stay in the subgoal direction while in "Follow" mode, ready to restart the guiding task from the place that requires the least effort to arrive at the original goal. These adaptive motions ensure that the robot guided the users' group along the shortest path. After the users' group returned to the robot, the robot again changed its direction of motion, and guided the users' group to the destination smoothly (point C to D in Fig. 4.19 (c)). It was observed that the robot prepared to restart the guiding task while in "Follow" mode by always standing in the direction of the goal, and the robot took the users to their destination via the shortest path by updating its path online.

4.4.3 Guiding efficiency

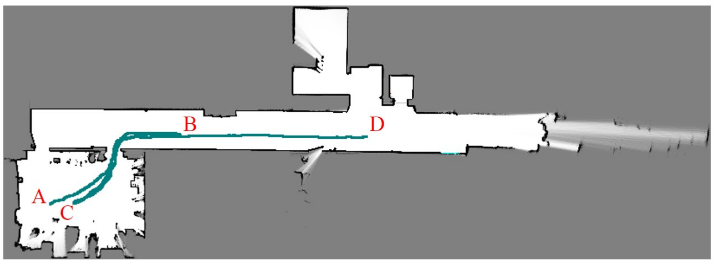
The proposed framework also improved the guiding efficiency of the robot. Here, the guiding efficiency is defined as the inversely proportional function of the trajectory length. If the users' group takes the same root to move towards the destination, the shorter trajectory length means the better the guide efficiency. The proposed method is compared with the conventional method in which the robot keeps the fixed distance with the users. The guiding trajectories are shown in Fig. 4.20 when the robot guides an cooperative users' group. It is observed that the trajectories are almost same (trajectory by the proposed method shown in Fig. 4.20 (a) and trajectory by the conventional method shown in Fig. 4.20 (b)) when the robot guides a



(a) Guiding the users' group from A to B



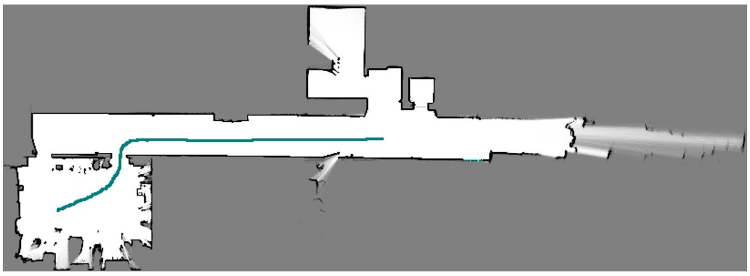
(b) following the users' group from B to C



(c) Guiding the users' group from C to D

Figure 4.19: Motion trajectories of the robot when guiding an uncooperative users' group who walked around the room.

cooperative users' group. It proves that the proposed method is comparable with the conventional method even if the users are cooperative. However, the trajectories have a big difference when the users are uncooperative. The guiding trajectories are shown in Fig. 4.21 when the robot move in totally same way. To ensure that the motion of users' group are same, the relative distance between the user and the robot is recorded and applied. It is observed that the trajectory of the proposed method is much less than


(a) Guiding by the proposed method

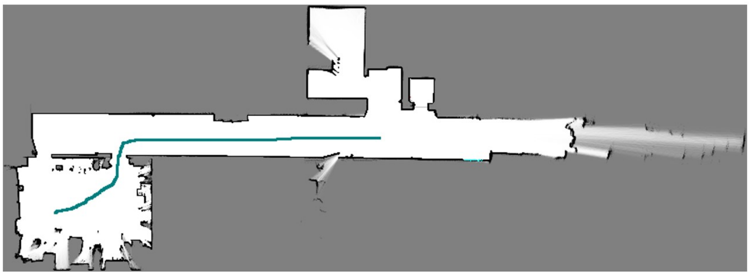

(b) Guiding by the conventional method

Figure 4.20: Motion trajectories of the robot when guiding the cooperative users' group by proposed and conventional methods.

the conventional method (trajectory by the proposed method shown in Fig. 4.21 (a) and trajectory by the conventional method shown in Fig. 4.21 (b)). This is caused by the robot controlled by the proposed method not only followed the users under the "follow" mode, but also considered the direction of the goal and prepared to change back to the "guide" mode in the best posture. Whenever the users want to return to the original path, the robot can smoothly restart the guide task.

All experimental videos were uploaded to the Internet: *www.youtube.com/watch?v=6V-13O7All8&feature=youtu.be*

4.5 Conclusion

In this chapter, a framework for adaptively controlling the motion of a guide robot to accompany users and provide sociable services to them is proposed.

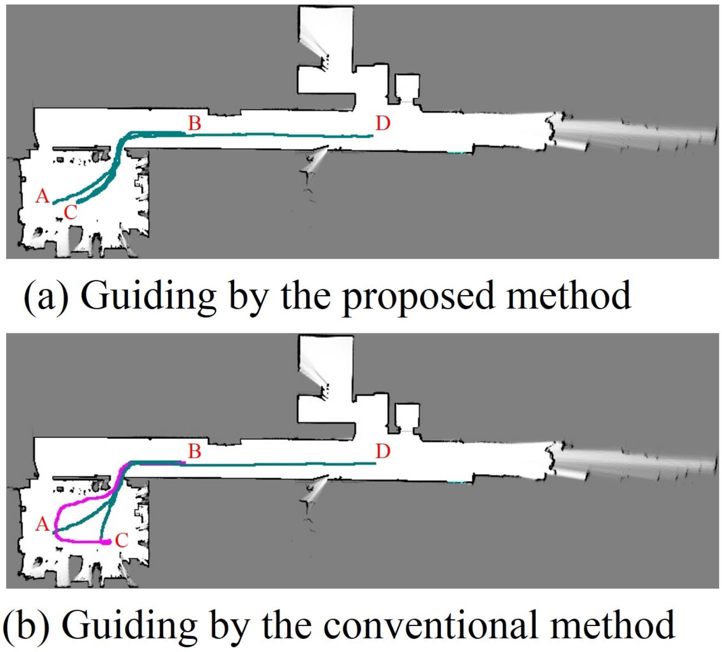


Figure 4.21: Motion trajectories of the robot when guiding the uncooperative users' group by proposed and conventional methods.

Different from the conventional guide robots [67, 69, 70], the guide robot controlled by the proposed framework did not give any restrains to the users, but adapted to the motions of the users. By setting special potential fields for the users and subgoal separately, and integrated them with other basic potential fields to control the robot, the robot guided the users considerately. Instead of being defined by multiple rules of motion, adaptive robot motions were generated naturally and automatically using the proposed framework. The robot can adjust its speed to meet the intent of the users and alternate between "Guide" and "Follow" modes to maintain the guiding relationship. Under "Follow" mode, the robot will always prepare to return to "Guide" mode by staying at the best position in the subgoal direction. Moreover, the shortest path is updated online to ensure that the users are guided along the shortest path. By comparing the simulation and experimental results, the usefulness of the proposed framework was confirmed.

On the other hand, the proposal's feasibility must be further inspected, because experimental environments and the number of the users were limited. Future work will focus on applying and verifying the proposed framework for guiding tasks in public places, where the environment is more complex and many kinds of tours with different numbers of people must be guided in a sociable way. The robust recognition and tracking of particular users in a crowded environment should also be studied. The proposed framework exclusively concerns sociable guiding services from the perspective of adaptive motion, to ensure that the guide robot can always accompany with the users. Other perspectives can also be considered in the future work such as sound or posture-guiding services using a humanoid robot, sound instructions to the environment and carrying the luggage for the users while accompanying with them.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this thesis, a framework is designed to control the robot to move adaptively in a socially acceptable way. Different from the conventional works that focused on realizing the guide task, this work tried to improve the quality of the guide task, allowing the users move more freely without any restrains. The sociable guide robot should not ask the users to follow it and maintain a proper distance with it. Instead, the robot should understand the will of the users and adapt to their motions.

Firstly, the conventional studies about mapping and path planning for mobile robot have been surveyed. The 2D map is generated by the immobile area grid map based SLAM method. By combining the 3D information from Kinect sensor with the 2D information from LRF sensor, the generated map reflected the 3D obstacles like desks and chairs in the space, and deleted potential mobile objects like human beings. By generating and updating the shortest path, and controlling the robots by potential field method, the robot was guided by the global optimum path successfully under dynamic environment.

Secondly, a general system to simultaneously recognize and track multiple people is proposed. The 3D distance information is used for detecting the positions of the users and they are tracked by using particle filter method. Here, the users are recognized by their faces and color information, which are registered to the system in advance, and the recognition results are used as the observation likelihood for the particle filter. The users' group formed by different numbers of users under the dynamic environment is successfully tracked by this method. Moreover, this tracking method is applied to more complex environment by tracking the children behaviors in the classroom of a nursery school, where the children were playing together. People moving in indoor environment with good illuminations have been continuously tracked more than 30 minutes, and the tracking error is never diverged. The results proved that the particular people can be continuously recognized and tracked even under crowed environment with the proposed system. The recognition algorithm keeps recognizing the detected people to improve the tracking accuracy, and the tracking algorithm records the motion history of the people and predict their positions to improve the recognition accuracy. Compared with conventional methods [49], the system has higher tracking accuracy and it can automatically modify the errors that caused by occlusions. That is to say, even if the system failed to track the right person, it will be modified when the particular user is recognized again.

Thirdly, a framework for adaptive motion control of the robot is proposed. Different from the conventional guide robots [67, 69, 70], the guide robot controlled by the proposed framework did not give any restrains to the users, but adapted to the motions of the users. By setting special potential fields for the users and subgoal separately, and integrating them with other basic potential fields to control the robot, the robot guided the users considerately. Instead of being defined by multiple rules of motion, adaptive robot motions were generated naturally and automatically using the proposed framework. With the single framework, the guide robot achieved (1) adjust its speed to match that of the user (e.g., young people may move faster than elderly, or the same group of people may change their speed depending on their interests); (2) maintain its relationship by maintaining the social distance with the users (e.g., by following them if they deviate from the guiding path); (3) prepare to restart the guiding task during "Follow" mode; and (4) take the users to their destination by the shortest path.

5.2 Future work

For this thesis, the simultaneous people recognition and tracking system is evaluated under lab environments. Although the tracking system is successfully applied to the nursery school for tracking the children behaviors, it is not sure that the system can work well under public places, like museums or airports. Future works need to collect real tour information and evaluate the system by tracking the users in public places.

Meanwhile, the adaptive motion control framework for the guide robot realized sociable guidance by accompanying the users. However, some users may hope the robot to wait in a place when they want to walk around for a while, or the robot needs to judge the situations by itself to decide whether follow the users when they deviate the original path or wait for them. More sociable adaptive motions needs to be developed in the future.

Moreover, the adaptive motion control is realized for the sociable guide robot in this thesis, but other social designs are also needed to the guide robot before real application. For example, the robot needs to be designed more like a humanoid robot and more acceptable to the users. The robot also needs to interact with users more properly by adding the speech recognition system and generating proper gestures. The robot should also be able to intend the mental states of the users for proving proper services.

References

[1] Sherine E.,"Effective Rapport in Tourist Guiding (Interpretation of Themes)," Journal of Socialomics, vol. 5, no. 172, pp. 1-5, 2016.

[2] Zenaida L. C.,"Principles and Ethics of Tour Guiding," Rex Bookstore, Inc., 2008.

[3] Hanqin Q. Zhang and Ivy Chow,"Application of importance-performance model in tour guides' performance: evidence from mainland Chinese outbound visitors in Hong Kong," Tourism Management, vol. 25, no. 1, pp. 81-91, 2004.

[4] Chang, Kuo-Chien,"Examining the effect of tour guide performance, tourist trust, tourist satisfaction, and flow experience on tourists' shopping behavior," Asia Pacific Journal of Tourism Research 19.2 (2014): 219-247.

[5] Al-Wazzan, Asraa, et al,"Tour-guide robot," 2016 International Conference on Industrial Informatics and Computer Systems (CIICS), 2016.

[6] Burgard, Wolfram, et al,"The interactive museum tour-guide robot," Aaai/iaai. 1998.

[7] Burgard, Wolfram, et al,"Experiences with an interactive museum tour-guide robot," Artificial intelligence, vol. 114, pp. 3-55, 1999.

[8] Thrun, Sebastian, et al,"MINERVA: A second-generation museum tour-guide robot," Proceedings of IEEE international conference on Robotics and automation, vol. 3, pp. 1995-2005, 1999.

[9] Kanda, Takayuki, et al,"An affective guide robot in a shopping mal," Proceedings of the 4th ACM/IEEE international conference on Human robot interaction. ACM, 2009.

[10] Sakagami, Yoshiaki, et al,"The intelligent ASIMO: System overview and integration," IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, pp. 2478-2483, 2002.

[11] http://pc.watch.impress.co.jp/docs/news/550598.html

[12] http://www.toyota.co.jp/en/news/07/0822.html

[13] http://www.ubergizmo.com/2011/12/tour-guide-robot-to-boost-tourist-arrivals/

[14] http://edition.cnn.com/2013/11/20/tech/innovation/skycall-the-drone-tour-guide/

[15] Bicchi A, Marigo A, Piccoli B, "On the reachability of quantized control systems," IEEE Transactions on Automatic Control, vol. 47, pp. 546-563, 2002.

[16] Bei G, Computational Geometry, "Algorithms and Applications," Tsinghua University Press, 2005.

[17] Khatib O, "Real-time obstacle avoidance for manipulators and robots," Proceedings of IEEE International Conference on Robotics and Automation, pp. 500-505, 1986.

[18] Ratering S, Gini M, "Robot navigation in a known environment with unknown moving obstacles" Autonomous Robots, vol. 1(2) pp.149-165, 1995.

[19] Podsedkowski L, Nowakowski J, Idzikowski M, et al, "A new solution for path planning in partially known or unknown environment for nonholonomic mobile robots," Robotics and Autonomous Systems, vol. 34, pp. 145-152, 2001.

[20] H. Durrant-Whyte and T. Bailey, "Simultaneous localisation andmapping (SLAM): Part I," IEEE Robotics and Automation Magazine, vol.13, no. 2, pp. 99-110, 2006.

[21] T. Bailey and H. Durrant-Whyte, "Simultaneous localisation andmapping (SLAM): Part II," IEEE Robotics and Automation Magazine, vol.13, no. 3, pp. 108-117, 2006.

[22] Y. Liu, J. Dong and F. Sun, "An efficient navigation strategy for mobile robots with uncertainty estimation," 7th World Congress on Intelligent Control and Automation, pp. 5174-5179, 2008.

[23] C.-C. Wang, C.E. Thorpe, S. Thrun, M. Hebert, and H.F. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," I. J. Robotic Res., Vol.26, No.9, pp. 889-916, 2007.

[24] M. Liu, S. Huan and G. Dissanayake, "Feature based SLAM using laser sensor data with maximized information usage," IEEE International Conference on Robotics and Automation, pp. 1181-1186, 2011.

[25] P. Besl and N. McKay, "A method for registration of 3-D shapes," IEEE trans. Pattern Analysis and Machine Intelligence, vol. 14, no. 2, pp. 239-256, 1992.

[26] D. M. Cole and P. M. Newman, "Using laser range data for 3D SLAM in outdoor environments," IEEE International Conference on Robotics and Automation, pp. 1556-1563, 2006.

[27] R. Halterman and M. Bruch, "Velodyne HDL-64E LIDAR for unmanned surface vehicle obstacle detection," Proc. International Society for Optical Engineering, vol. 7692, pp. 224-231, 2010.

[28] J. Welle, D. Schulz, T. Bachran and A. Cremers, "Optimization techniques for laser-based 3D particle filter SLAM," IEEE International Conference on Robotics and Automation, pp. 3525-3530, 2010.

[29] J. Weingarten and R. Siegwart, "EKF-based 3D SLAM for structured environment reconstruction," IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3834-3839, 2005.

[30] K. Lin, C. Chang, A. Dopfer and C. Wang, "Mapping and localization in 3D environments using a 2D laser scanner and a stereo camera," Journal of Information Science and Engineering 28, pp. 131-144, 2012.

[31] D. Wolf and G. Sukhatme, "Mobile robot simultaneous localization and mapping in dynamic environments," IEEE International Conference in Robotics and Automation, vol.19, no.1, pp.53-65, 2005.

[32] D. Hahnel, W. Burgard, D. Fox and S. Thrun, "An efficient fastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," IEEE/RSJ International Conference on Robotics and Systems, vol. 1, pp. 206-221, 2003.

[33] S. Yang and C..Wang, "Feasibility grids for localization and mapping in crowded urban scenes," IEEE International Conference on Robotics and Automation, pp. 2322-2328, 2011.

[34] D. Holz, C. Lorken and H. Surmann, "Continuous 3D sensing for navigation and SLAM in cluttered and dynamic environments," 11th International Conference on Information Fusion, pp. 1-7, 2008.

[35] A. Ito, K. Takahashi, and M. Kaneko, "Robust mapping for mobile robot based on immobile area grid map considering potential moving objects," IEEJ Transactions on Electronics, Information and Systems, vol. 134, no. 2, pp. 192-204, 2014.

[36] R. Kurazume, H. Yamada, K. Murakami, Y. Iwashita and T. Hasegawa, "Target tracking using sir and mcmc particle filters by multiple cameras and laser range finders," Proc. 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS08), pp. 3838-3844, 2008

[37] Munaro, Matteo, Filippo Basso, and Emanuele Menegatti, "Tracking people within groups with RGB-D data," 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2012.

[38] R. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," IEEE International Conference on Robotics and Automation, pp. 1-4, 2011.

[39] D. Hahnel, D. Schulz, andW. Burgard, "Mobile robot mapping in populated environments," Advanced Robotics, Vol.17, No.7, pp.579-598, 2003

[40] T.-D. Vu, J. Burlet, and O. Aycard, "Grid-based localization and local mapping with moving object detection and tracking," Information Fusion, Vol.12, No.1, pp.58-69, 2011.

[41] N.C. Mitsou and C.S. Tzafestas, "Temporal occupancy grid for mobile robot dynamic environment mapping," Control Automation, 2007. MED'07. Mediterranean Conference on, pp.1-8, 2007.

[42] S.-W. Yang and C.-C.Wang, "Feasibility grids for localization and mapping in crowded urban scenes," 2011 IEEE International Conference on Robotics and Automation (ICRA), pp.2322-2328, 2011.

[43] T. Moeslund, A. Hilton, and V. Kruger, "A survey of advances in vision based human motion capture and analysis," Computer Vision and Image Understanding, vol. 104, no. 2, pp. 90-126, 2006.

[44] A. Ali and K. Terada, "A framework for human tracking using Kalman filter and fast mean shift algorithms," Int. Conf. Computer Vision Workshop, pp. 1028-1033, 2009.

[45] S. Tang, M. Andriluka, A. Milan, K. Schindler, S. Roth and B. Schiele, "Learning people detectors for tracking in crowded scenes," Inter. Conf. Computer Vision, pp. 1049-1056, 2013.

[46] M. Andriluka, S.Roth, B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1-8, 2008.

[47] D. Glas, T. Miyashita, H. Ishiguro, and N. Hagita, “Laser-based tracking of human position and orientation using parametric shapemodeling,” Advanced Robotics, vol. 23, no. 4, pp. 405-428, 2009.

[48] C. Chou, J. Li, M. Chang and L. Fu, “ Multi-robot cooperation based human tracking system using Laser Range Finder,” IEEE International Conference on Robotics and Automation, pp. 532-537, 2011.

[49] D. Glas, T. Miyashita, H. Ishiguro, and N. Hagita, “Simultaneous people tracking and localization for social robots using external laser range finders,” IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 846-853, 2009.

[50] N. Kirchner, A. Alempijevic, and A.Virgona, “Head-to-shoulder signature for person recognition,” Proc. IEEE Int. Conf. Robotics and Automation, pp. 1226-1231, 2012.

[51] D. Hansen, M. Hansen, M. Kirschmeyer, R. Larsen, and D. Silvestre, “Cluster tracking with time-of-flight cameras,” IEEE Conf. Computer Vision and Pattern Recognition Workshop, pp. 1-6, 2008.

[52] D. Brscic, T. Kanda, T. Ikeda, and T. Miyashita, “Person tracking in large public spaces using 3-D range sensors,” IEEE Trans. on Human-Machine Systems, vol. 43, no. 6, 2013.

[53] R. Sivalingam, A. Cherian, J. Fasching, N. Walczak, N. Bird, V. Morellas, B. Murphy, K. Cullen, K. Lim, G. Sapiro and N. Papanikolopoulos, “A multi-sensor visual tracking system for behavior monitoring of at-risk children,” IEEE International Conference on Robotics and Automation, pp. 1345-1350, 2012.

[54] W. Choi, C. Pantofaru, and S. Savarese, “A general framework for tracking multiple people from a moving camera,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 35, no. 7, pp. 1157-1591, 2013.

[55] L. He, Y. Chao and K. Suzuki, “A run-based two-scan labeling algorithm,” IEEE Transactions on Image Processing, vol. 17, no. 5, pp. 749-756, 2008.

[56] Takigawa, E., S. Hosoi, and M. Kawade, “Gender, age, human race identification and estimation technology of facial attributes-OKAO Vision,” Image Lab, 15.4, 2004.

[57] Masahiro Shiomi and Norihiro Hagita, “Preliminary investigation of supporting childcare at an intelligent playroom,” The 2nd Int. Conf. on Human Agent Interaction, 2014.

[58] I. Hwang, Jang H., Nachman L., and Song J., “Exploring inter-child behavioral relativity in a shared social environment: a field study in a kindergarten,” Proceedings of the 12th ACM International Conference on Ubiquitous Computing, 2010, pp. 271-280.

[59] M. Srivastava, Muntz R., and Potkonjak M., “Smart kindergarten: sensor-based wireless networks for smart developmental problem solving environments,” The 7th Annual International Conference on Mobile Computing and Networking, 2001, pp. 132-138.

[60] M. Shiomi and N. Hagita, “Social acceptance of a childcare support robot system,” 24th IEEE International Symposium on Robot and Human Interactive Communication, pp. 13-18, 2015.

[61] Asoh H., Motomura Y., Asano F., Hara I., Hayamizu S., Itou K., Kurita T., Matsui T., Vlassis N., Bunschoten R. and Krose B., “Jijo-2: an office robot that communicates and learns,” IEEE Intelligent Systems, vol. 16, no. 5, pp. 46-55, 2001.

[62] Stuckler J., Holz D. and Behnke S., “Demonstrating everyday manipulation skills in RoboCup@Home,” IEEE Robotics and Automation Magazine, vol. 10, no. 2, pp. 34-42, 2012.

[63] Tzou Jyh-Hwa and Su Kuo L., “The development of the restaurant service mobile robot with a Laser positioning system,” Proceedings of the 27th Chinese Control Conference, pp. 662-666, 2008.

[64] Mamun K.A., Sharma A., Hoque A.S.M. and Szecsi, T., “Remote patient physical condition monitoring service module for iWARD hospital

robots, " Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE), pp. 1-8, 2014.

[65] Byung-Ok Han, Young-Ho Kim, Kyusung Cho and Yang, H.S., "Museum tour guide robot with augmented reality," International Conference on Virtual Systems and Multimedia (VSMM), pp. 223-229, 2010.

[66] Mizobuchi Y., Wang S., Kawata K. and Yamamoto M. : "Motion control of guide robots tracking relative distance and running speed considered the characteristic of a guided person by speech dialogue", The 44th General Conference of JSME Chuogoku-Shikoku Branch, vol. 2006(44), pp. 419-420, 2006.

[67] Fleury S., Alami R., Chatila R., Bailly G., Brethes L., Cottret M., Danes P., Dollat X., Elisei F., Ferrane I., Herrb M., Infantes G., Lemaire C., Lerasle F., Manhes J., Marcoul P., Menezes P., and Montreuil V., "Rackham: An interactive robot-guide," The 15th IEEE International Symposium on Robot and Human Interactive Communication, pp. 502-509, 2006.

[68] M. Shiomi, T. Kanda, H. Ishiguro and N. Hagita, "A larger audience, please!: encouraging people to listen to a guide robot," Proc. HRI2010. pp. 31-38, 2010.

[69] Oyama T., Yoshida E., Kobayashi Y. and Kuno Y., "Tracking visitors with sensor poles for robot's museum guide tour," The 6th International Conference on Human System Interaction (HSI), pp. 645-650, 2013.

[70] Pandey A.K. and Alami R., "A step towards a sociable robot guide which monitors and adapts to the person's activities," International Conference on Advanced Robotics (ICAR), pp. 1-8, 2009.

[71] Nakazawa K., Takahashi K. and Kaneko M., "Movement Control of Accompanying Robot Based on Artificial Potential Field Adapted to Dynamic Environments," IEEJ Transactions on Electronics, Information and Systems, vol.134, no. 2, pp. 293-302, 2014.

[72] Dijkstra E.W., “A note on two problems in connexion with graphs”, Numerische Mathematik 1, pp.269-271, 1959.

[73] Edward T. Hall, “The Hidden Dimension”, Doubleday Company, 1966.

[74] Costa W.F., Matsuura J.P., Santana, F.S. and Saraiva A.M. : “Evaluation of an ICP based algorithm for simultaneous localization and mapping using a 3D simulated P3DX robot”, IEEE Conference on Robotics Symposium and Intelligent Robotic Meeting (LARS), pp. 103-108, 2010.

[75] Berzuini, C., Best, N.G., Gilks, W., Larizza, C., “Dynamic conditional independence models and Markov chain Monte Carlo methods,” Journal of the American Statistical Association, vol. 92, no. 440, pp.1403-1412, 1997.

[76] Zhang B., Nakamura T., Abe K., Attamimi M., Nagai T., Omori T., Oka N. and Kaneko M., “Children behavior tracking and personal identification by multiple Kinect sensors,” The 30th Annual Conference of the Japanese Society for Artificial Intelligence, 4K4-1, 2016.

[77] Zhang B., Nakamura T. and Kaneko M., “Tracking method of multiple users group in a crowd for autonomous guiding robot,” The 77th National Convention of IPSJ, 4C-03, pp.2-5～2-6, 2015.

[78] Zhang B., Nakamura T. and Kaneko M., “Robust tracking of multiple particular users for group guiding robot,” The 32th Annual Conference of the Robot Society of Japan, 3I3-03, 2014.

List of Publications

Journal papers:

[1] B. Zhang, T. Nakamura, R. Ushiogi, T. Nagai, K. Abe, T. Omori, N. Oka and M. Kaneko, "Simultaneous children recognition and tracking for childcare assisting system," Journal of Signal and Information Processing, vol.7, no.3, pp.148–159, Aug. 2016. (Related to the contents of Chapter 3)

[2] B. Zhang, T. Nakamura, and M. Kaneko, "A framework for adaptive motion control of autonomous sociable guide robot," IEEJ Transactions on Electrical and Electronic Engineering, vol.11, no.6, pp.786–795, Aug. 2016. (Related to the contents of Chapter 4)

International Conference Papers:

[1] B. Zhang, T. Nakamura and M. Kaneko, "Adaptive fusion of multi-information based human identification for autonomous mobile robot," The 24th IEEE International Symposium on Robot and Human Interactive Communication, (RO-MAN 2015), IS02-poster, Sep. 2015. (Related to the contents of Chapter 4)

[2] B. Zhang, T. Nakamura, R. Ushiogi, T. Nagai, K. Abe, T. Omori, N. Oka and M. Kaneko, "Robust children behavior tracking for childcare assisting robot by using multiple Kinect sensors," The 8th International Conference on Social Robotics, Nov. 2016. (Related to the contents of Chapter 3)

Other Publications and Presentations:

[1] B. Zhang, T. Nakamura, K. Abe, M. Attamimi, T. Nagai, T. Omori, N. Oka and M. Kaneko, “Children behavior tracking and personal identification by multiple Kinect sensors,” The 30th Annual Conference of the JSAI, 4K4-1, Jun. 2016.

[2] 張斌, 中村友昭, 阿部香澄, アッタミミムハンマド, 潮木玲奈, 長井隆行, 大森隆司, 岡夏樹, 金子正秀, “保育支援システムのための Kinect を用いた子どもの行動追跡,” SSI2015, GS13-5, Nov. 2015.

[3] B. Zhang, T. Nakamura, and M. Kaneko, “A considerate guidance framework of servicing a users’ group for autonomous guiding robot,” 電子情報通信学会, 信学技報, vol. 115, no. 283, CNR2015-15, pp. 41-46, Oct. 2015.

[4] B. Zhang, T. Nakamura, and M. Kaneko, “Tracking Method of Multiple Users Group in a Crowd for Autonomous Guiding Robot,” 情報処理学会第 77 回全国大会, 4C-03, pp.2-5～2-6, Mar. 2015.

[5] B. Zhang, T. Nakamura, and M. Kaneko, “Robust recognition of multiple particular users’ group for autonomous guiding robot,” 電子情報通信学会 2015 年総合大会, D-12-69, Mar. 2015.

[6] B. Zhang, T. Nakamura and M. Kaneko, “Robust tracking of multiple particular users for group guiding robot,” 第 32 回日本ロボット学会学術講演会, 3I3-03, Sep. 2014.

[7] B. Zhang and M. Kaneko, “Efficient Human-robot Interaction for Searching Unknown User under Complex Environment with Multiple People and Objects,” 日本感性工学会生命ソフトウェア部会, 生命ソフトウェアシンポジウム 2013 講演論文集, G3-2, pp.35-39, Oct. 2013.

[8] B. Zhang and M. Kaneko, “Efficient search for receiver based on audio-visual information under complex environment - Toward an intelligent delivery robot -,” 2013 年映像情報メディア学会年次大会, 2-6, Aug. 2013.

[9] 張斌, 高橋桂太, 金子正秀, “効率的なインタラクションを用いた複雑環境下でのユーザ探索,” ROBOMEC2013, 2P1-P15, May 2013.

Acknowledgements

I would like to give my deepest thanks to my supervisor, Prof. Masahide Kaneko, for his generously supporting and guiding me through my master and doctor courses. When I came to Japan in 2011, I had no experience about research and I had some problems about living in Japan as I could not speak Japanese at that time. Prof. Kaneko helped me to apply the MEXT scholarship, and taught me from the very beginning to do my research. He also gave me a lot of suggestions about my daily life and job hunting. Without Prof. Kaneko's help, I could not get my degrees and work in Japan. And also, I am specially thankful to Dr. Nakamura, for his very meaningful suggestions to my research.

I would like to express my special thanks to all the members of the supervisory committee: Prof. Kazuo Tanaka, Prof. Hiroshi Yokoi, Prof. Takayuki Nagai and Prof. Masafumi Uchida.

Also, I would like to thank all the members in my lab for their helps on my research and daily life. I really enjoyed my time studying in this lab. The special experiences of living and studying with them will be the most important memories in my life.

Finally, I would like to thank all my families and my friends. Especially for my parents, they encouraged and supported me all the time no matter what kind of life I chose for. I also appreciate the other family members and my friends who encouraged me to work harder.

I am grateful to all the people who helped me during this time.

Author Biography

Bin Zhang was born in Tangshan City, Hebei Province, China, on March 12, 1989. He received his B.E. degree from Harbin Engineering University, China in 2011 and M.E. degree from the University of Electro-Communications, Japan in 2013. He is currently a doctoral student in the Department of Mechanical Engineering and Intelligent Systems, the University of Electro-Communications, Japan. His research interests include robot vision, autonomous mobile robots, and human-robot interaction.

Mr. Zhang is a student member of the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan and the Robot Society of Japan.